

BEST AVAILABLE COPY

G06F 15/72

(72)Inventor : **SETO KUNIO**

2005/11/11 14:59

[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's
decision of rejection]
[Date of requesting appeal against
examiner's decision of rejection]
[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

11/11/2005
14:59:00
2005/11/11
14:59:00
2005/11/11
14:59:00

11/11/2005
14:59:00
2005/11/11
14:59:00
2005/11/11
14:59:00

11/11/2005
14:59:00
2005/11/11
14:59:00
2005/11/11
14:59:00

11/11/2005
14:59:00
2005/11/11
14:59:00
2005/11/11
14:59:00

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A)

平4-93894

⑬ Int. Cl.⁵

識別記号

庁内整理番号

⑭ 公開 平成4年(1992)3月26日

G 09 G 5/26
G 06 F 15/72

3 5 5 U

8320-5G
8125-5L

審査請求 未請求 請求項の数 2 (全31頁)

⑮ 発明の名称 文字処理方法および装置

⑯ 特 願 平2-206621

⑰ 出 願 平2(1990)8月3日

⑱ 発 明 者 瀬 戸 邦 雄 東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

⑲ 出 願 人 キヤノン株式会社 東京都大田区下丸子3丁目30番2号

⑳ 代 理 人 弁理士 丸 島 儀 一 外1名

明 細 書

1. 発明の名称

文字処理方法および装置

2. 特許請求の範囲

(1) 標準的な図形情報として用意された閉輪郭座標情報と、これに含まれる複数の閉輪郭を区別するための輪郭情報を、指定された出力サイズに応じて拡大または縮小演算した後、ドット情報に変換し、出力する文字処理装置が用いるビットマップ展開用文字データに変換する文字処理方法であって、

外輪郭、内輪郭を判別する手段と、

水平線、垂直線を抽出する手段と、

これを分類した後、水平線、垂直線の要素組を決定する手段と、

分類する手段と、

水平、垂直スペース領域を設定する手段と、

低画素時省略順位を設定する手段とを具備し、

当該文字の輪郭を表現するための制御点座標と、

複数の閉輪郭を区別する輪郭情報と、

バンド情報と、

前述バンド情報を参照するための制御点情報と、
前述バンド情報に含まれない制御点の座標値情報からなるビットマップ展開用文字データに変換することを特徴とする

文字処理方法。

(2) 標準的な図形情報として用意された閉輪郭座標情報と、これに含まれる複数の閉輪郭を区別するための輪郭情報を、指定された出力サイズに応じて拡大または縮小演算した後、ドット情報に変換し、出力する文字処理装置が用いるビットマップ展開用文字データに変換するに際し、
標準的な図形情報を格納する格納手段と、
読み出し手段と、

外輪郭、内輪郭を判別する手段と、

水平線、垂直線を抽出する手段と、

これを分類した後、水平線、垂直線の要素組を決定する手段と、

分類する手段と、

水平、垂直スペース領域を設定する手段と、

低画素時省略順位を設定する手段とを具備し、
当該文字の輪郭を表現するための制御点座標と、
複数の閉輪郭を区別する輪郭情報と、
バンド情報と、

前述バンド情報を参照するための制御点情報と、
前述バンド情報に含まれない制御点の座標値情報
からなるビットマップ展開用文字データに変換す
る変換手段と、

これを格納する格納手段を

具備することを特徴とする

文字処理装置。

3. 発明の詳細な説明

〔産業上の利用分野〕

本発明は、標準的な図形情報として用意された閉輪郭座標情報と、これに含まれる複数の閉輪郭を区別するための輪郭情報を、指定された出力サイズに応じて拡大または縮小演算した後、ドット情報に変換し、出力する文字処理装置が用いるビットマップ展開用文字データに変換する文字処理方法および装置に関する。

認できる (W H S I W Y G = W h a t Y o u S e e I s W h a t Y o u G e t) 必要がある。

これらの要求を満たすための一解決手段として、文字画像の輪郭点定義による文字発生方式（以後アウトラインフォント方式と称す）を採用するシステムがふえてきた。

アウトラインフォント方式によれば、例えば文字の輪郭を表現するための制御点座標データをシステム中のROM (Read Only Memory) などに予め用意し、文字コードインデックスに応じて一時記憶装置RAM (Random Access Memory) に読み出して一時格納し、さらに指定された文字出力サイズと出力すべきプリンタなどの機器解像度に応じて、前記制御点座標データを数学的に拡大または縮小演算した後、ドット情報に変換することにより、文字出力サイズや出力すべきプリンタなどの機器解像度に対して自由度の高い文字発生が可能である。

本方式は従来のドットフォントによる文字発生方

〔従来の技術〕

マイクロコンピュータの価格性能比の向上、小型高精細のページプリンタや高精細モニタディスプレイの一般化といったハードウェアの進歩、これに文書処理ソフトウェアの充実が伴って、近年電子的手法による印刷物（または印刷版下）作成システムが急速に一般化している。

このようなシステムにおいて、印刷物の最も重要な要素である文字画像出力技術にも以下のような進歩が要求されている。

第1に、より広範囲のサイズでより美しく文字を再現する必要がある。

第2に、より自由に訴求力のあるレイアウトを実現するため、文字の自由な展開方法や斜体、長・平体のような変形、シャドウ付け、ハッチング付け（内部タイリング）といった修飾などをおこなう必要がある。

第3に、前記システムにおいて、指定したレイアウトが最終出力であるプリンタなどに対するイメージとして、例えばモニタディスプレイ上で直接確

式と比較した場合、特に大きなサイズの文字を美しく表現可能であることがよく知られている。

〔発明が解決しようとしている課題〕

（従来技術の問題点）

しかしながら、アウトラインフォント方式は1960年代前半に、非常に高価なマイクロフィルム出力システムや大規模な印刷版下出力システムに適用するものとして開発されたものである。

つまり、これは例えば1,000 DPI (Dot Per Inch、1インチあたりの解像ドット数を表わす) 以上の超高解像度出力機用に意図されたものであり、高解像度出力を満たすための文字発生用データの容量削減を第一目的としていた。

1,000 DPIにおいては1文字当りの画素数（理論的に割り当てられるドットの総数）が通常の本文用として一般的な活字10ポイント相当においておよそ140×140である。

これを、ドットフォントによる文字発生方式を用いた場合、1ドットを1ビットに割り当てると1文

字あたり容量は2,450バイトも必要であるが、アウトラインフォント方式によれば、1/5から1/10に圧縮可能である。

一方、前述超高解像度出力機にアウトラインフォント方式を用いてビットマップ文字を出力する場合、前記制御点座標値を数学的に拡大または縮小演算した結果の量子化誤差が大きな問題となることは少なかった。

つまり、文字画像再現技術としての精度や再現時の美しさが問題になることは少なかった。

しかし前述の電子的手法による印刷物作成システムはこのような大規模システムのハードウェア構成とはかなり異なり、個人で使えるワークステーションをベースに成り立つことを前提としている。従って、ハードウェアは可能な限りの低価格において性能を発揮しなければならない。

つまり、前述の小型高精細のページプリンタは例えば、B4サイズ400DPIのレーザプリンタであり、その場合、前述の超高解像度出力機と比較すると文字画像を再現するに際し、1文字当り

式を採用する必要があるが、モニタディスプレイの持ちうる解像度は前述したプリンタの解像度よりさらに低い場合が多く、例えば100DPI程度である。

100DPIモニタディスプレイの場合、10ポイントサイズ1文字当りの画素数は14×14、8ポイントサイズでは11×11である。

これに対し、漢字をドット表現するに最低必要な画素数は、16×16：字種を判別可能（複雑な文字では字面の省略が必要）、24×24：明朝体とゴシック体を最低判別可能、であることが良く知られている。

このことから、100DPI程度のモニタディスプレイを用いてアウトラインフォント方式による文字表現をおこなうことは、本文用文字サイズにおいて画素数の面で非常に不利であった。

つまりこの場合、出力結果のビットマップ文字の美しさの損失は第1の問題点で述べた以上のものであり、前述のWYSIWYGの要求に対し、実用上の大きな障害となっていた。

に割り当てられる画素数が少ない。

例えば、400DPIプリンタの場合、10ポイントサイズ1文字当りの画素数は56×56、8ポイントサイズでは44×44である。

前述の量子化誤差の影響は出力機器の解像度が低いほど顕著となる。

よって第1の問題点は、前述の電子的手法による印刷物作成システムにおいて用いられるページプリンタなどの解像度においては、従来のアウトラインフォント方式では文字を構成する線の必要以上の太まり、線巾のばらつき、線間のつぶれ、曲線のがたつきや、図形的対称性の損失などが著しく、結果として出力されるビットマップ文字の美しさが大きく損なわれることである。

特に本文用として使用頻度の高い文字サイズについてその欠点が多い。

さらなる第2の問題点は、文書処理ソフトウェアの充実が要求するWYSIWYGを実現するためには、プリンタ、モニタディスプレイなど接続される機器の解像度に依存しない共通の文字出力方

このため、従来のシステムでは文字処理方法または方式をモニタディスプレイ用、プリンタ用など接続される出力機器の解像度に応じ別々に備える、もしくは、1文字当りの画素数の大小により別々に備えることが一般的であった。

しかしこのようなシステムにおいては、書体の増加など当該システムの発展的変更に対応して各々のフォントを同時に変更しなければならず、これを怠った場合、モニタディスプレイにおいて表示された文字とプリンタに出力された文字が異なるなどの問題があった。

また、モニタディスプレイにおいてアウトラインフォント方式を持っていないシステムではドットフォントにより擬似的な拡大、縮小などをおこない表示するシステムもあるが、レイアウトに大切な最終イメージの確認は不完全なものとなってしまう。

以上、WYSIWYGの要求を満たすため、アウトラインフォント方式を、低解像度のモニタディスプレイも含め、接続される機器の解像度に依存

しない共通の文字出力方式とするには、従来、技術的に大きな問題があり、またこれを補うために用いられる従来の解決方法にも前述の如く、種々の問題点があった。

また、さらなる第3の問題点は、前述の問題点を解決するために、前記制御点座標値を数学的に拡大または縮小演算した結果の量子化誤差を、審美的に吸収するための強制的な補正を行なう矯正データを付加するにも、各文字に対する矯正データ追加のための人的作業が膨大となるばかりでなく、従来のアウトラインフォント方式による文字データの資産を生かせないなどの重大な問題点があった。

〔課題を解決するための手段〕

本発明によれば、標準的な図形情報として用意された閉輪郭座標情報と、これに含まれる複数の閉輪郭を区別するための輪郭情報を、指定された出力サイズに応じて拡大または縮小演算した後、ドット情報に変換し、出力する文字処理装置が用いるビットマップ展開用文字データに変換するする

ブロック図の一例である。

なお、本発明を実施するにあたって必要な装置はこれに限る必要はなく、単体の機器であっても、また複数の機器の組み合わせによるシステムであっても実施可能であることは言うまでもない。あるいは、装置にプログラムを供給することにより、本発明の機能が達成されることを含むことは言うまでもない。

第1図において、

1は、後述する標準となる文字データを格納する格納領域、

2は、後述するビットマップ変換用文字データを格納する格納領域、

3および4は、後述する本発明を実行するためのプログラムを格納するプログラム格納領域、

5は、後述する本発明の文字処理装置に対する入力データをあらかじめ格納してあるドキュメントファイル格納領域、

6は、本実施例におけるシステムをオペレータなどが操作するに際し用いるキーボード装置、

に際し、

外輪郭、内輪郭を判別する手段と、

水平線、垂直線を抽出する手段と、

これを分類した後、水平線、垂直線の要素組を決定する手段と、

分類する手段と、

水平、垂直スペース領域を設定する手段と、

低画素時省略順位を設定する手段とを具備し、

当該文字の輪郭を表現するための制御点座標と、複数の閉輪郭を区別する輪郭情報と、

バンド情報と、

前述バンド情報を参照するための制御点情報と、

前述バンド情報に含まれない制御点の座標値情報からなるビットマップ展開用文字データに変換することにより、

広範囲のサイズで、高品位の文字を、出力可能なビットマップ展開用文字データを効率良く得ることが可能としたものである。

〔実施例〕

第1図は、本発明を実施するに必要なシステムの

7および8は、後述する本発明を実行するためのプログラムがそれら各々の実行に際し、ロードされるに必要なプログラム用RAM（ランダムアクセスメモリ）、

9は、本システムが作動するに際し、必要とされる処理上の中間的な全てのデータを一時的に記憶するためのデータ用RAM、

10、11、12、および13は、いずれも後述するビットマップで構成される21で示すフレームバッファであり、各々実際の出力機器である14、15、16、および17で示すモニタディスプレイCRT1、CRT2、およびプリンタPRT1、PRT2に文字を表示または出力するために接続されているRAM、

18は、本システムが作動するに際し、その構成要素を制御するCPU（中央演算装置）、

19は、システムのハードウェアとして考えた場合、例えば、前述1、2、3、4、および5をファイルとして格納するディスク装置、

20は、システムのハードウェアとして考えた場

合、ごく一般的なDRAM（ダイナミックランダムアクセスメモリ）であってもよい。

22は、システムの機器構成などを記憶してあるROM（リードオンリメモリ）である。

なお、14、15、16、および17で示した複数のモニタディスプレイやプリンタはそれらの機器が持つ出力解像度が異なっている。

例えば、14で示すCRT1は100DPI、15で示すCRT2は72DPI、16で示すPRT1は400DPI、17で示すPRT2は1000DPIと仮定する。

標準となる文字データの定義に関する説明

本実施例における標準となる文字データの定義を説明するに当たり、先ず前述文字データを標準的な図形情報として考えた場合の輪郭座標情報に関する説明を以下におこなう。

また以下に説明する標準となる文字データの作成行程については、一般的なまたは専用のCAD（コ

つまり、以下に述べるフォント座標系、文字輪郭制御点データ、および輪郭情報である。

フォント座標系

第2図に本発明におけるフォント座標系の一例を示す。

第2図では、X軸およびY軸で定義される2次元座標系を左上原点O（x000, y000）とし、右下ボディ対角点Z（x999, y999）とする。

一般にx000=y000=0、x999=y999=(512~1024)などが実際の値として都合がよい。

上記OとZを対角とする矩形は活字のボディに相当するものである。

また、原点の位置やX軸およびY軸の方向については、適応されるシステムに都合のよい方法でかわらない。

ンピュータ支援によるデザイン）システムなどを用いて、多角形図形や曲線図形をその座標点データとして入力する方法や、もしくは文字原稿をデジタル走査することにより一旦ビットマップイメージでコンピュータ内に取り込んだ後、輪郭追跡のアルゴリズムを用いたソフトウェアにより特徴点を抽出する方法など種々の方法により実現可能であるが、本発明ではその方法について述べるものではない。

以下、日本語で用いる漢字を中心に説明するが、例えば欧文のアルファベットなど他言語に用いる文字図形であっても同様であることは言うまでもなく、一般図形（ベクタグラフィックス）を扱う情報についても同様に本発明が適用可能である。標準となる文字データはこれを1文字単位で読み出すための、本説明においては省略するインデックスコードと共に、第1図1に示すFDT1に格納される。

個々の標準となる文字データは以下の3種類の情報から構成される。

文字輪郭制御点データ

上述フォント座標系における標準となる文字データの一例として、第3図（A）に示す明朝体の「機」と、これに対応する文字輪郭制御点のデータを第4図に示す。

文字輪郭制御点のデータは第4図（A）に示す文字輪郭制御点（以後、制御点と称す）配列、および同じく（B）に示すこれらに関連づけるための、後述する輪郭開始点情報（以後、輪郭情報と称す）からなる。

本例の場合、制御点総数は157であり、各々をp0~p156と記する。

また各制御点PはX座標とY座標をもち、p0~p156にたいし、実際の各座標値を第4図（A）に示す如く、（x0, y0）~（x156, y156）と記する。

さらに、各制御点Pが文字の直線部端点を示すか、後述する曲線補間のための制御点を示すかを区別するための属性情報Cをもち、各制御点Pに対応

し、第4図に示す如く、 $c_0 \sim c_{156}$ と記する。
 $c_0 \sim c_{156}$ は、例えば各制御点Pが直線部端点の場合ST、さらに、曲線補間のための制御点の場合はCS(曲線開始点)、CM(曲線中間点)、またはCT(曲線終了点)などを実際の値としてもつ。

ST、CS、CM、およびCTは実際の値として例えば、各々0、1、2、および4などとして格納される。

第3図においてはST、CS、およびCT、を示す制御点を黒点で、CMを示す制御点を白点で表わす。

前述、曲線補間のための曲線補間式については本例では多重化ノット3次Bスプライン曲線または3次Bezier曲線を用いる。

上記曲線の数学的な性質、および図形表現についての応用例については、例えば共立出版社「bit」Vol. 13, No. 10~12に詳しいので、本実施例においてはその説明を省略する。

曲線補間式は上記曲線を含め、それ以外のいかな

即ち、配列として隣接した制御点が本来閉じた輪郭として接続されるべきか、また別の輪郭上の制御点であるかを区別するための情報であり、本実施例では以下の様な情報として持つ。

第3図(A)で示す文字図形の場合、閉じた輪郭の数を表わす輪郭開始点総数は10であり、第4図(B)に示す如く各々を $t_0 \sim t_9$ と記する。本実施例における輪郭情報は、後述する処理の便宜上、輪郭開始点と輪郭終了点を持つ(実際には上記の輪郭開始点もしくは輪郭終了点のいずれかのみを持てばよい)。

つまり第1番目の輪郭情報 t_0 は輪郭開始点 p_0 と輪郭終了点 p_{118} を、第2番目の輪郭情報 t_1 は輪郭開始点 p_{119} と輪郭終了点 p_{124} をもつ。第3番目以降は上記と同様である。

さらに、輪郭制御点総数No. of Cpと輪郭開始点総数ntを輪郭情報先頭にもち、本例の場合、第3図(A)および第4図から明らかなように $\text{No. of Cp} = 157$ 、 $\text{nt} = 10$ を実際の値として持つ。

る曲線補間方法でも構わないが、本実施例で上記3次曲線を用いているのは以下の理由による。

つまり、第3図(B)、(C)、(D)、および(E)に示すような欧文書体で用いられる文字、例えばローマン書体やサンセリフ書体など一般的欧文書体の「R」、「O」などをはじめとする殆どの文字、数字、さらに、「(」:丸カッコ、「O」:円、などをはじめとするシンメトリックな曲線を持つ記号図形などについて、曲線部分を表わす制御点の配置を直線部分を表わす制御点の配置と同様に扱うことが可能であることによる。

即ち、上記のような取扱いが可能な曲線補間式であればいかなる曲線でも構わない。

例えば、2次Bスプライン曲線や2次Bezier曲線も上記条件を満たすことが周知である。

輪郭情報

上述制御点において、隣接した制御点同士の連結関係を示す情報を輪郭情報としてもつ。

以上が標準となる文字データ定義に関する説明であり、前述の従来技術によるアウトラインフォント方式における一般的な文字図形定義である。

しかしながら、これを用いてビットマップ変換処理を実行するに際し、前記第4図に示した制御点の座標値を単純に拡大または縮小演算するだけでは、前述の電子的手法による印刷物作成システムにおいては既に説明したように種々の問題点が生じる。

よって、本発明によれば以下に説明する手段を用いて、標準となる文字データよりビットマップ変換用文字データを自動的に作成する。

上記手段は以下に示すステップを踏む。即ち、標準となる文字データの読み出し、

外輪郭、内輪郭の判別、

水平線バンド情報の作成、

垂直線バンド情報の作成、

低画素時省略順位の設定、

ビットマップ変換用文字データの格納

である。

これを第5図フローチャートで示す。

また、当該手段は第1図3に示すPRG1にソフトウェアプログラムとして格納されており、6に示すKBDからの指令により7に示すPRAM1へロードされ、18に示すCPUにより逐次制御されるものである。

また、標準となる文字データよりビットマップ変換用文字データを自動的に作成するにあたり、必要とする全ての中間的なデータはいずれも第1図9に示すDARAMに一時格納される。

標準となる文字データの読み出し

標準となる文字データを第1図1で示すFD T1より前述インデックス情報を元に個々の文字単位で読み出し、第1図9に示すDARAMに一時格納する。

外輪郭、内輪郭の判別

において、さらにX座標最小値をもつ制御点 $p_j(X_{min}, Y_{min})$ を抽出する。

p_j は各閉輪郭に対し、1個のみに限定される。上記ステップ(6.1)およびステップ(6.2)に関しては、第7図(A)にその簡単な説明図を示す。

ステップ(6.3)

以下に説明するステップ(6.4)にて必要とする「横切り」カウント変数TRGを初期化する($TRG = 0$)。

ステップ(6.4)

以下に説明する「横切り」のカウントをおこなうことにより外輪郭、内輪郭の判定をおこなう。

即ち、全ての輪郭ベクタ(1つの閉輪郭に属し、2個の隣接した制御点で形成される直線ベクタ $p_n \cdot p_{n+1}$)についてそのY成分変位が上述 Y_{min} を含み、そのベクタの Y_{min} におけるX成分が上述 X_{min} よりも小なる場合、TRGを1増加させる($TRG = TRG + 1$)。

これは上述 $p_j(X_{min}, Y_{min})$ を目的座

上述の如く、文字輪郭制御点データは各々の輪郭開始点から輪郭終了点までの1連の制御点で1個、または2個以上の閉輪郭を形成しているが、あるものはその内部を「黒」とみなし、あるものはその内部を「白」として区別する必要がある。

例えば、制御点進行方向右側を常に「黒」とするよう配列を順序付ける必要がある。

これは以下のように自動的に判別可能である。

第6図に外輪郭、内輪郭判別処理のフローチャートを、また第7図にその概念的補足図を示し、これらに従って以下に説明する。

個々の閉輪郭に対し、第8図に示すステップ(6.1)～(6.5)の手順の処理をおこない、外輪郭、内輪郭の判別をおこなう。

ステップ(6.1)

各閉輪郭に属する制御点において、Y座標最小値をもつ制御点 $p_i(X, Y_{min})$ を抽出する。ただし、 p_i は1個以上の複数個が存在しうる。

ステップ(6.2)

上記1個以上の複数個が存在しうる制御点 p_i に

模値として、Y座標 Y_{min} についてボディ左端 $x000$ から水平方向に走査したとき、いくつかの輪郭ベクタを横切るかをカウントすることになる。この結果、変数TRGの値が0を含む偶数の場合、外輪郭であり、また奇数の場合、内輪郭であることが判定できる。

ステップ(6.5)

上述ステップ(6.4)による判定にもとずき、制御点進行方向右側を常に「黒」とするよう配列を順序付けるためには、第7図(B)および(C)に示すように、 p_n の両隣接点 p_{n-1} と p_{n+1} について p_n となす角度の大小比較をおこなう。即ち、外輪郭と判定された場合、第7図(B)に示す如く、制御点進行方向を角度大なる隣接点が p_n の次に配列するよう再配列する。

また、内輪郭と判定された場合、第7図(C)に示す如く、角度小なる隣接点が p_n の次に配列するよう再配列する。

上記ステップ(6.1)からステップ(6.5)を全ての閉輪郭について判定する。

上述した処理の結果、第7図(D)に示す如く外輪郭を表わす制御点配列は時計方向に進み、内輪郭を表わす制御点配列は反時計方向に進む。これを以後、CW (Clockwise) ルール (時計方向をCW、反時計方向をCCW) と称す。

水平線バンド情報および垂直線バンド情報の作成

第5図フローチャートで示す水平線バンド情報の作成および垂直線バンド情報の作成は以下のステップをふむ。

即ち、

水平〔垂直〕線ベクタの抽出と分類、
水平〔垂直〕線分類 (CLASS) のソート、
水平〔垂直〕線要素組の決定、
水平〔垂直〕線バンド分類、
水平〔垂直〕線バンド0の決定、
水平〔垂直〕線バンド1以降の決定、
水平〔垂直〕線バンド分類のソート、
水平〔垂直〕線バンドのボディ全体に対する適応

たは垂直線を表わす輪郭制御点座標値と同様に扱うことができる。

換言すれば、本処理は上記文字図形を定義する輪郭制御点の水平線要素および垂直線要素に着目するものである。

また本実施例では、水平線要素と垂直線要素を別々に管理し、第5図フローチャートに示すとおり先ず第1に水平線バンド情報の作成処理をおこない、しかる後、垂直線バンド情報の作成処理をおこなうよう実施される。

しかし、その両者は主に上記処理における演算の対象となる座標成分 (本実施例の場合、X座標成分とY座標成分) が異なるのみであり、全く同様の処理を実行することができる。

以下、水平線要素についてバンド情報の作成処理を第8図および第11図(A)、(B)で示すフローチャートの各ステップに沿って説明する。垂直線要素については、上述理由によりその説明を省略し、処理の結果を後述することとする。

水平〔垂直〕スペース領域の設定、である。

これを第8図フローチャートで示す。

バンド情報についての概念的な説明

以下、第8図フローチャートにそって、上述水平線バンド情報および垂直線バンド情報の作成について説明するが、それに先立ち、まずその概念的な説明をおこなう。

本処理は以下に説明する如くの、電子的手法による印刷物作成システムで用いられる、最も一般的な印刷用書体がもつ審美的デザイン思想に基づくものである。

即ち、明朝体やゴシック体などに代表される多くの上述印刷用書体の場合、それらの字形要素の50%以上は水平線または垂直線で構成されている。また本実施例の場合、第3図を用いて既に説明した如く、欧文書体やシンメトリックな曲線を持つ記号文字の曲線部輪郭制御点座標値は、水平線ま

水平線バンド情報の作成処理

第6図を用い説明した如く、CWルールに基づいて配列順序を決定した (外輪郭、内輪郭の判別の結果にもとずき再配列された) 制御点の各隣接する2点で構成されるベクタ ($p_k \cdot p_{k+1}$) (k = 各閉輪郭の輪郭開始点 ~ 輪郭終了点、但し p_k = 輪郭終了点の場合、 p_{k+1} = 輪郭開始点) のX変移 dx とY変移 dy 、および

X座標値 $p(x)$ とY座標値 $p(y)$ について、以下の処理をおこなう。

ステップ (8.1)

水平線ベクタの抽出と分類

$dy = 0$ なる ($p_k \cdot p_{k+1}$) の組を抽出する。即ち、 $p_k(y) = p_{k+1}(y)$ なる ($p_k \cdot p_{k+1}$) の組を抽出する。

次に、 dx の符号 (正または負、0はあり得ない) が等しく (以後、方向分類と称す)、Y座標 $p(y)$ の値が等しいものを同一の水平線分類 (以後、CLASSと称す) とする。

このとき、前述CWルールによりdx符号が正の分類は当該文字データの水平線上辺を表わし、dx符号が負の分類は水平線下辺を表わすことが第7図(D)より理解されるであろう。

ステップ(8. 2)

水平線分類(CLASS)のソート

上記水平線分類(CLASS)の各々について、上述方向分類別にY座標p(y)値の昇順に水平線分類(CLASS)をソートする。

さらに、各水平線分類におけるX座標値の最小値Lminおよび最大値Lmaxを求める。

ステップ(8. 3)

水平線要素組の決定

dx符号が正である水平線分類(CLASS)、すなわち上辺に相当する、の各々について、Y座標p(y)の値の昇順に、以下の方法で水平線要素組として対応するdx符号が負の水平線分類(CLASS)、すなわち下辺に相当する、を決定する(対応づける)。

決定の対象となる分類(CLASS)の値(Y座

標値)をCLASS0、

LminをLmin0、

LmaxをLmax0、そして

水平線要素組として対応すべき分類(CLASS)

の値(Y座標)をCLASS1、

LminをLmin1、

LmaxをLmax1とすると、

以下の判定、即ち、

CLASS0 < CLASS1、かつ

Lmin1 < Lmax0、かつ

Lmin0 < Lmax1、かつ

(CLASS1 - CLASS0)の値が最小である水平線分類を抽出する。

上記の判定により、水平線要素の対応する上辺と下辺の組合せによる水平線要素組が決定する(対応づけられる)。

第9図(A)には第3図(A)で示した「種」の水平線要素組の決定結果の例を、また第9図(B)には同様に垂直線要素組の決定結果の例を示す。また、これを概念的により理解しやすくするため、

第10図に文字図形上での反映例を示す。

ステップ(8. 4)

水平線バンド分類

ステップ(8. 3)にて、決定された水平線要素組の各々について、X座標の最小値Eminと最大値Emaxを求める。

ステップ(8. 5)

水平線バンド0の決定

本処理の初期化として、CLASS最小値をもつ水平線要素組を水平線バンド0として分類する。この水平線要素組の(Emin, Emax)を水平線バンド0に対応する(Zmin, Zmax)とする。

水平線要素組の各々について、以下の条件で判定をおこなう。即ち、

$Emin < Zmax$ 、かつ、 $Zmin < Emax$

上記判定結果が「真」であった場合、

以下の処理をおこなう。

その水平線要素組をバンド0に分類する。

Zminを(Zmin, Emin)の最小値で置

き換える。

Zmaxを(Zmax, Emax)の最大値で置き換える。

ステップ(8. 6)

水平線バンド1以降の決定

上記ステップ(8. 5)の条件判定結果が「偽」であった場合、

新たなバンド(既決のバンドn: $n = 0 \sim$ 、に対しバンドn+1)として分類する。

バンドn+1に対する初期化として、この線要素組の(Emin, Emax)を各々バンドn+1に対応する(Zmin, Zmax)とする。

これを全ての水平線要素組について判定し、上記の如く既決のバンドnに分類されない場合には、新たなバンドn+1を作成する。

ステップ(8. 7)

水平線バンド分類のソート

後述する処理において、バンドを当該文字のボディ全体に適用するため、本ステップに至る処理の結果作成されたバンド分類をZminの昇順に並

べ換える。

第3図(A)で示した「種」の水平線バンドについて、具体的な結果を第12図に示すが、

水平線バンド0として $Z_{min} = x107$ 、 $Z_{max} = x84$ が格納され、

水平線バンド1として $Z_{Xmin} = x69$ 、 $Z_{Xmax} = x62$ が格納される。

ここに、 $x107 < x84 < x69 < x62$ である。

ステップ(8.8)

水平線バンドのボディ全体に対する適応

本ステップに至る処理において、いずれのバンドにも分類されなかった部分が存在するが、これらを既決バンドに割り当てる処理をおこなう。

そのため、例えば以下のような処理をおこなう。
バンド0の Z_{min} の値を $x000$ とする(ボディ最左端まで拡張する)。

最終バンドの Z_{max} の値を $x999$ とする(ボディ最右端まで拡張する)。

バンド0以外のバンドの Z_{min} の値を直前のバ

$z1p4$ 、 $z1p5$ 、 $z1p6$ 、および $z1p7$ に対し、

水平スペース領域 $zsp0$ 、 $zsp1$ 、 $zsp2$ 、 $zsp3$ 、 $zsp4$ 、 $zsp5$ 、 $zsp6$ 、 $zsp7$ 、および $zsp8$

が設定される。

ただし、各々の線要素組は対応する分類の値を含むが、

スペース領域については $y000$ と $y999$ を除き対応する分類の値を含まない。

第3図(A)に示す「種」の水平線バンドBAND1について具体的な結果を以下に示す。

線要素組の値は

$y17 \leq z1p0 \leq y119$ 、
 $y24 \leq z1p1 \leq y125$ 、
 $y126 \leq z1p2 \leq y133$ 、
 $y134 \leq z1p3 \leq y94$ 、
 $y36 \leq z1p4 \leq y141$ 、
 $y142 \leq z1p5 \leq y149$ 、
 $y150 \leq z1p6 \leq y46$ 、

ンドの $(Z_{max} + 1)$ とする。

ステップ(8.9)

水平スペース領域の設定

各バンドに対して、線要素組以外の字形要素およびスペース領域(ボディの地の部分、即ち、第3図(A)に示す文字図形を通常の文字として考えた場合、文字の画線部でない部分)に対する領域(即ち、スペース領域とはボディの領域より線要素組の占める領域を除いた部分を示す)を以下の方法で設定する。

分類の先頭に $y000$ (ボディ最上端値)を挿入する。

分類の最終に $y999$ (ボディ最下端値)を挿入する。

しかる後、分類の先頭から2個ずつを組とし、スペース領域 $zsp0$ 、 $zsp1$ 、...として設定する。

第3図(A)に示す「種」の水平線バンドBAND1について具体的な結果を例にとると、

水平線要素組 $z1p0$ 、 $z1p1$ 、 $z1p2$ 、 $z1p3$ 、

$y50 \leq z1p7 \leq y82$

である。

スペース領域の値は

$y000 \leq zsp0 < y17$ 、
 $y119 < zsp1 < y24$ 、
 $y125 < zsp2 < y126$ 、
 $y133 < zsp3 < y134$ 、
 $y34 < zsp4 < y36$ 、
 $y141 < zsp5 < y142$ 、
 $y149 < zsp6 < y150$ 、
 $y46 < zsp7 < y50$ 、
 $y82 < zsp8 \leq y999$

つまり、スペース領域の値は

$y000 \leq zsp0 \leq y17 - 1$ 、
 $y119 + 1 \leq zsp1 \leq y24 - 1$ 、
 $y125 + 1 \leq zsp2 \leq y126 - 1$ 、
 $y133 + 1 \leq zsp3 \leq y134 - 1$ 、
 $y34 + 1 \leq zsp4 \leq y36 - 1$ 、
 $y141 + 1 \leq zsp5 \leq y142 - 1$ 、
 $y149 + 1 \leq zsp6 \leq y150 - 1$ 、

$y 4 6 + 1 \leq z s p 7 \leq y 5 0 - 1$ 、
 $y 6 2 + 1 \leq z s p 8 \leq y 9 9$
 である。

低画素時省略順位の設定

以下に詳述する理由により、第5図フローチャートにおける低画素時省略順位（以下、省略順位と称す）の設定をおこなう。

低画素時省略順位の設定についての概念的説明

第3図（A）に示す「亀」における水平線バンド BAND 1 を具体例として説明する。

水平線バンド BAND 1 には水平線要素組 8 個、および水平スペース領域 9 個（水平線要素組の個数 + 1）が存在する。

つまり、第3図（A）で示す「亀」の「亀」の部分（一般に「つくり」と称される）をビットマップとして、再現（または表現）する場合、Y 方向

レータの認識可能な範囲で最小限にとどめるため、以下の方法で低画素時省略順位（以下、省略順位と称す）を設定する。

また第~~4~~¹⁶図にピクセルサイズが 17 以下の場合の「亀」部ピクセル割り当て方法を示すが、以下に説明する処理はその出力として本図のような結果を得るためにおこなうものである。

第11図フローチャートでその詳細を示す上述省略順位の設定は以下のステップをふむ。

即ち、

初期設定（省略順位変数 ZEP の初期化）、

最優先省略領域の設定、

第二優先省略領域の設定、

第三優先以降の省略領域を順次設定、

対応する線要素組に対しての省略順位設定、

である。

以下、第11図フローチャートにそって、低画素時省略順位の設定について説明する。

ステップ（11. 1）

初期設定

に 17（水平線要素組の個数 + 水平スペース領域の個数）ピクセルの画素数（これを個々の水平線バンドにおける最低再現画素数 MINPIX と称す）があれば、各水平線要素を 1 ピクセルの間隔を置いてピクセル配置可能である。

これは逆に、その文字を出力するに際し、割り当てられる Y 方向の画素数が 16 ピクセル以下の場合、必ず何らかの字形の「崩れ」、または線間の「つぶれ」が生ずることを意味する。

一方、前述の如く、100DPI の CRT を用い、一般的な本文用文字サイズである 10 ポイント文字サイズを WYSIWYG 表示する場合、1 文字に割り当てられるピクセルサイズは 14（= 13.89）、12 ポイントでは同様に、17（= 16.67）である。

つまり、前述「亀」の部分は上記 100DPI の CRT では 14 ポイントより小さなポイントサイズの場合、字形の崩れが不可避である。

よって、上記問題を解決するための一手段として、上述字形の崩れが不可避の場合においても、オベ

省略順位変数 ZEP を初期化する

（ZEP = MINPIX）。

ステップ（11. 2）

最優先省略領域の設定

最上端水平スペース領域、および最下端水平スペース領域に対し、水平線要素（以下、単に線要素と称す）以外の字形要素が存在するか否かを判定し、その結果により以下の処理をおこなう。

線要素以外の字形要素について、両者とも存在する、または両者とも存在しない場合、

スペース領域（対応する分類の差分）が小さいほうに ZEP 値をセットし、ZEP 値を 1 減ずる（ZEP = ZEP - 1）。

しかる後、スペース領域が大きいほうに対して ZEP 値をセットし、ZEP 値を 1 減ずる（ZEP = ZEP - 1）。

両者のうちいずれかにのみ存在する場合、

存在しないほうに ZEP 値をセットし、ZEP 値を 1 減ずる（ZEP = ZEP - 1）。

しかる後、存在するほうに対して ZEP 値をセッ

とし、ZEP値を1減ずる ($ZEP = ZEP - 1$)。

ステップ (11. 3)

第二優先省略領域の設定

次に比較をするスペース領域を文字ボディ最中心位置に移動させる。

文字ボディ最中心位置のスペース領域は1個の場合と2個の場合がある(「機」の水平線バンドBAND1の場合、文字ボディ最中心位置の水平スペース領域はzsp4が1個のみである)。

スペース領域が1個のみの場合、

該当するスペース領域にZEP値をセットし、ZEP値を2減ずる ($ZEP = ZEP - 2$)。

スペース領域が2個の場合、

該当するスペース領域(対応する分類の差分)が小さいほうにZEP値をセットし、ZEP値を2減ずる ($ZEP = ZEP - 2$)。

しかる後、スペース領域が大きいほうに対してZEP値をセットし、ZEP値を2減ずる ($ZEP = ZEP - 2$)。

ステップ (11. 4)

位が設定されていないが、その省略順位値ZEPは常に1であり、これをセットする。

以上のステップを各バンドについて実行し、全てのスペース領域の省略順位を設定する。

以上の結果を、「機」の水平線バンドについて第12図(A)に、同様に垂直線バンドについて第12図(B)に示す。

ビットマップ変換用文字データの格納

以上のようなステップで座標要素毎に作成されたバンド情報を付加し、後述する格納形式に従って、標準となる文字データの場合と同様に、1文字単位で読み出すための後述するフォントインデックスと共に第1図2に示すFDT2に格納する。

ビットマップ変換用文字データの格納形式について、第~~14~~¹⁴図と共に以下に説明する。

第~~14~~¹⁴図(A)で示すように、ビットマップ変換用文字データの格納形式は大別して、輪郭情報CONT-INF、オフセット情報OFFSET-INF、

第三優先以降の省略領域を順次設定

以後比較をするスペース領域を順次文字ボディ外側方向に移動させ、以下の様に全てのスペース領域にZEP値をセットする。

該当するスペース領域(対応する分類の差分)が小さいほうにZEP値をセットし、ZEP値を2減ずる ($ZEP = ZEP - 2$)。

しかる後、スペース領域が大きいほうに対して同様にZEP値をセットし、ZEP値を2減ずる ($ZEP = ZEP - 2$)。

ステップ (11. 5)

対応する線要素組に省略順位を設定する。

各スペース領域とその直後(直下)で隣接する線要素組をグループ化し、各線要素組に対しグループ化されたスペース領域と同一の省略順位を設定する。

但し、先頭(最上端)のスペース領域と最終(最下端)のスペース領域はグループ化の対象としない。

この結果、先頭の線要素組はこの段階では省略順

X座標情報X-INF、Y座標情報Y-INFから構成される。

輪郭情報(CONT-INF)

輪郭情報はその文字の輪郭の数と各々の輪郭終了点を示す制御点配列番号の配列から構成される。

オフセット情報(OFFSET-INF)

オフセット情報は後述する各座標データのデータオフセットを示す4個のオフセットポイントで、X-BAND-INF-OFF、X-CORD-INF-OFF、Y-BAND-INF-OFF、Y-CORD-INF-OFFから構成される。X-BAND-INF-OFFは後述するX座標のバンド情報先頭オフセットを示すオフセットポイントである。

X-CORD-INF-OFFは後述するX座標の制御点情報先頭オフセットを示すオフセットポ

インタである。

$Y_BAND_INF_OFF$ は後述するY座標のバンド情報先頭オフセットを示すオフセットポインタである。

$Y_CORD_INF_OFF$ は後述するY座標の制御点情報先頭オフセットを示すオフセットポインタである。

座標情報 (X_INF 、 Y_INF)

座標情報に関してはX座標情報 X_INF とY座標情報 Y_INF が独立しているが、各々は同一の構成からなり、主にプログラム上の座標成分を変えるだけで全く同一の処理で扱うことができる。座標情報は各々バンド情報 $BAND_INF$ と制御点情報 $CORD_INF$ から構成される。前述の如く、X座標情報とY座標情報は同一の構成なので、以下にY座標情報を例にとって、説明する。

る。

制御点情報 ($CORD_INF$)

¹⁴
第~~14~~図 (C) に $CORD_INF$ の構造を示す。
¹⁵
また、第~~15~~図に「種」を例とした具体的な制御点情報を部分的に示す。

制御点情報はその文字を表現する各々の制御点に対し、バンド番号 $BAND_NO$ 、クラス番号 $LINE_SEQ$ 、属性フラグ REG_F を持つ。

属性フラグ REG_F

属性フラグ REG_F は、後述説明する出力機器解像度に合致させた制御点情報の作成処理を、高品位かつ効率よくおこなうための付加情報であり、コピー属性 CPY 、クラス属性 ZLO 、 ZLC 、 ZLM 、 ZSM 、補正属性 ADG 、曲線属性 $CV C$ をその内部に記録する。

バンド情報 ($BAND_INF$)

¹⁴
第~~14~~図 (B) で示すように、バンド情報 $BAND_INF$ は1個以上のバンドを定義している。 $MINPIX$ は各バンドで判別された最低再現画素の最大値を格納している。 No_of_Band と $Band_Offset$ は各々複数のバンド情報を扱うための情報の個数とオフセットポインタである。

バンド ($BAND$)

個々のバンドは可変長である。 No_of_Line は個々のバンドに存在する線要素の個数を格納する。 $ZLPO_Array$ は ZEP_Array は低画素時省略順位であり($No_of_Line + 1$)個の配列である。 $ZLPC_Array$ は $CLASS$ 値を格納する($No_of_Line \times 2 + 1$)個の配列であ

属性フラグ REG_F の説明を、同様に水平線要素について以下に述べる。

コピー属性 CPY

直前の制御点と同じ座標値を持つことを示す。つまり、前述ステップ(8.1)にて説明した $dy = 0$ なる($p_k \cdot p_{k+1}$)を抽出した結果の制御点 p_{k+1} に対し、本属性を付与する。

クラス属性 ZLO

線要素基準値を持つことを示す。つまり、前述 $CLASS$ 配列における dx 符号が正である($p_k \cdot p_{k+1}$)に対し、本属性を付与する。

第9図(A)、第12図(A)に例をとると、 $y17$ 、 $y24$ 、 $y6$ 、 $y126$ 、 $y134$ などである。

クラス属性 ZLC

線要素相対値を持つことを示す。つまり、前述 $CLASS$ 配列における dx 符号が負である($p_k \cdot p_{k+1}$)に対し、本属性を付与する。

第9図(A)、第12図(A)に例をとると、 $y119$ 、

y 1 2 5、y 8 4、y 1 3 3、y 2 4 などである。

クラス属性 Z L M

線要素基準値と線要素相対値には含まれた領域に存在することを示す。

つまり、制御点の座標値 $p_k(y)$ が (直前の Z L O 値) $< p_k(y) <$ (直前の Z L C 値) である p_k に対し、本属性を付与する。

クラス属性 Z S M

線要素相対値と線要素基準値には含まれた領域に存在することを示す。

つまり、制御点の座標値 $p_k(y)$ が (直前の Z L C 値) $< p_k(y) <$ (直前の Z L O 値) である p_k に対し、本属性を付与する。

補正属性 A D G

前述 C L A S S 配列に属さないが、 $dy = 0$ なる $(p_k \cdot p_{k+1})$ に対し、本属性を付与する。

曲線属性 C V C

前述説明した標準となる文字データの属性情報 C 属性情報 C と同様であり、各制御点 P が文字の直線部端点を示すか、後述する曲線補間のための制

に格納されるドキュメントファイル格納領域からのデータ受信部に相当する入力回路、21 は出力すべき画像データをビットデータとして記憶するフレームバッファ、31 は出力回路でありビデオ信号などのプリンタ用出力回路もしくは C R T 表示回路などに相当し、本回路によりプリンタや C R T など可視画像出力機器へ接続される。

なお第 ¹³~~14~~ 図に付記した上記番号は前述第 1 図の構成と一致させてある。

ただし、第 ¹³~~14~~ 図 30 で示す入力回路、および 31 で示す出力回路は第 1 図では省略されている。

フォントデータ

上述したビットマップ変換用文字データはこれらを複数文字の集合体として管理するため、以下に説明するフォントデータとして取り扱う。

即ち、フォントデータはフォントインデックス、コードインデックス、および個々のビットマップ変換用文字データから構成される。

御点を示すかを区別するための属性情報である。

ビットマップ文字の出力

第 ¹³~~14~~ 図は上記説明にもとずいて作成された、ビットマップ変換用文字データを、指定された出力機器の解像度と、指定された文字サイズに従って、ビットマップデータに変換し、出力する文字処理装置部分の構成図である。

また、第 ¹⁷~~18~~ 図は後述する文字発生部がビットマップ文字の出力に際し、作動する内容を説明するフローチャートである。

第 ¹³~~14~~ 図において、18 は C P U、4 は C P U およびシステムを制御する文字処理装置部のプログラムが格納された制御 R O M、32 は前述プログラムがビットマップ文字の出力に際し作動する文字発生部 C G、9 は R A M、2 はフォントデータが格納されたフォントメモリ、もしくは外部記憶装置等を含む R A M、30 は第 1 図 6 で示すキーボード、もしくは第 1 図 5 で示すディスク装置等

フォントインデックスは書体毎、文字セット (日本語文字セットの場合、J I S 第 1 水準、第 2 水準、システムに固有の文字種集合など) 毎の管理を取り扱うインデックス情報である。

また本実施例ではビットマップ変換用文字データのフォント座標系情報もフォント単位に管理している。

コードインデックスは上記の各フォント毎の文字コードと後述するビットマップ変換用文字データのデータ先頭およびデータ長を格納している。

ビットマップ変換用文字データは文字毎に第 ¹⁴~~15~~ 図および第 ¹⁵~~16~~ 図を用いて前述説明した如く、輪郭情報、オフセット情報、およびバンド情報、制御点情報からなる座標情報を格納している。

また、本システムの文字処理装置部分の制御手段はソフトウェアプログラムとして第 1 図 4 に示す P R G 2 に格納されており、6 に示す K B D からの指令により 8 に示す P R A M 2 へロードされ、18 に示す C P U により逐次制御されるものである。

また文字発生に際し、必要とする全ての中間的なデータはいずれも第1図9に示すD A R A Mに一時的格納される。

入力パラメータ

ある文字をプリンタやC R Tなど可視画像出力機器へ出力するに際し、入力回路からは以下のような文字出力コマンドとして指定される。

文字出力コマンド = {出力機器種別、文字サイズ、フォント種別、出力文字種}

出力機器種別は第1図14、15、16、および17の如くシステムに接続されているプリンタやC R Tなどの可視画像出力機器の解像度を指定するためのもので、第1図22に示すS Y S R O Mにあらかじめセットされている出力機器/解像度対応テーブルを参照することにより指定された出力機器の解像度を得る。

上記対応テーブルに格納されている実際の値は例えばD P I数である。

ると、

$$P I X _ S I Z E = P T _ S I Z E \times D _ D P I / 72$$

である。

P I X _ S I Z Eは出力機器におけるドットサイズであるため、式の右辺の結果は例えば四捨五入され整数値化される。

この結果、P I X _ S I Z Eの値は例えば、出力機器の解像度D _ D P I = 400とすると、文字サイズが32ポイントのとき178、16ポイントのとき89、12ポイントのとき67、10ポイントのとき56であり、D _ D P I = 100とすると、P I X _ S I Z Eの値は同様に44、22、17、および14である。

マスタフォントサイズ

指定されたフォント種別により、フォントインデックスから該当するフォントデータの後述するマスタフォントサイズを得て、P I X _ S I Z Eよ

文字サイズは指定された文字を出力するに際し、出力機器の解像度に依存しない一般的なサイズであることが好ましく、例えばポイントサイズ（本例では1ポイントを1/72インチとして説明する）等で指定される。

フォント種別は前述した書体や文字セットを指定するもので、これにより前述したフォントインデックスを参照する。

出力文字種は前述したコードインデックスを参照するための文字コードである。

出力ドットサイズの算出

入力回路から出力機器種別、文字サイズ、フォント種別が指定された後、まず指定された文字サイズが指定された出力機器の解像度において実際に展開されるべきドットサイズを算出する。

指定された文字サイズをP T _ S I Z E、指定された出力機器の解像度をD _ D P I、実際に展開されるべきドットサイズをP I X _ S I Z Eとす

り拡大、縮小率の算出をおこなう。

出力に際し、必要なフォント座標系情報は、例えば第2図を用い既に説明した左上原点Oの座標値と右下ボディ対角点Zの座標値である。

第2図のような座標系定義をした場合、左上原点Oの値(x 0 0 0, y 0 0 0)を(0, 0)としておけば、ビットマップ変換用文字データのサイズはZの値、即ち(x 9 9 9, y 9 9 9)である。

拡大、縮小率の算出

前述マスタフォントサイズをM A S T E R _ S I Z E、拡大縮小率をS C A L E _ Fとすると、

$$S C A L E _ F = P I X _ S I Z E / M A S T E R _ S I Z E$$

である。

S C A L E _ Fの値は、小数点以下の値が必要になる場合が多いので浮動小数点変数、または例えば10進法で小数点以下2桁以上の精度が保証されるようビットシフトを施した固定小数点変数と

して演算する。

出力機器解像度に合致させた制御点情報の作成

指定されたフォント種別および出力文字種により、フォントインデックスおよびコードインデックスを介してビットマップ変換用文字データを得た後、後述するドット展開回路に出力するための出力機器解像度に合致させた輪郭制御点情報（以後、出力解像度合致制御点と称す）を作成する。

ドット展開回路はなるべく単純な、例えばビットマップ塗り潰し回路であることが望ましく、前述した拡大縮小率 $SCALE_F$ を用いた演算を含め、指定された文字サイズにおいていかに美しく出力するかは全てこの出力解像度合致制御点の作成処理において制御する。

本処理も X 制御点情報と Y 制御点情報を独立かつ、同一の構成からおこなうことが可能であり、X/Y の区別は主に制御点情報の座標成分を変えるのみである。

解像度 400 DPI のプリンタに対し、32 ポイントサイズ以上の文字を出力する場合 ($PIX_SIZE = 32 \times 400 / 72 = 178$) などがこれに相当する。

バンドのメンバ、クラス配列の演算

各々の文字は 1 個以上のバンドをもつが、その個々について、以下の如くクラス配列の演算をおこなう。

線要素基準値の演算

線要素基準値の演算処理フローチャートを第¹⁸~~20~~図に示し、これに従って以下に説明する。

演算結果用配列 $D_ZLO[0:No_of_Line-1]$ を DARAM 上に確保する。

ここに、配列 $ARRAY[a:b]$ は配列要素が a から b までの配列要素をもつ $ARRAY$ という名前の配列を定義することを示す。但し a および b は整数で $a \leq b$ である。

$D_ZLO[i] =$
 $ROUND(ZLPO[i] \times SCALE_F)$
 $(i = 0 \sim No_of_Line - 1)$

以下に Y 座標情報を例にとって説明する。

前述した如く、ビットマップ変換用文字データは文字単位で前述した輪郭情報、オフセット情報、座標情報を格納しているが、第¹⁷~~18~~図フローチャートに示す如く、最低再現画素 $MINPIX$ ($MINPIX$ の値は個々の文字により異なる) と前述した実際に展開されるべきドットサイズ PIX_SIZE の比較結果を判別し、それらの条件に最も適した処理を選択する。

つまり、 PIX_SIZE が $MINPIX$ に対して十分大なる (例えば $PIX_SIZE \geq MINPIX \times 5$) 場合、 PIX_SIZE が $MINPIX$ 以下である場合、そしてその他の場合、というような条件によりそれぞれ別の制御点情報の作成をおこなう。

以下、上記条件の場合の各々について説明する。

PIX_SIZE が $MINPIX$ に対して十分大なる場合

$ZLPO[No_of_Line]$ の値はこの処理で必要としない。

ここに、 $ROUND(n)$ は変数 n を n の値に最も近い整数値と等しくさせる関数であり、例えば小数点以下を四捨五入して整数化する関数である。

線要素相対値の演算

線要素相対値の演算処理フローチャートを第 21 図に示し、これに従って以下に説明する。

演算結果用配列 $D_ZLC[0:No_of_Line-1]$ を DARAM 上に確保する。

$D_ZLC[i] = D_ZLO[i] +$
 $ROUND((ZLPC[j] - ZLPO[i]) \times SCALE_F)$

$(i = 0 \sim No_of_Line - 1,$
 $j = i + 1)$

$ZLPC[0]$ の値はこの処理で必要としない。

制御点情報配列より制御点座標値の作成

制御点情報配列より制御点座標値作成演算処理のフローチャートを第²¹~~22~~図に示し、これに従って以下に説明する。

出力解像度合致制御点の演算結果を格納するための配列 $D_CORD[0:No_of_Cp-1]$ を $DARAM$ 上に確保する。

線要素組に属さない制御点のためのオフセット座標値配列のポインタ $COFF_P$ をセットする。

REG_F の値が、

CPY のとき、

直前の制御点と同じ値を持つことを示す、即ち

$D_CORD[i] = D_CORD[i-1]$

ZLO のとき、

線要素基準値を持つことを示す、即ち

$D_CORD[i] = BAND[BAND_NO].$

$D_ZLO[LINE_SEQ]$

ZLC のとき、

線要素相対値を持つことを示す、即ち

$D_CORD[i] = BAND[BAND_NO].$

$D_ZLC[LINE_SEQ]$

ZLM のとき、

線要素基準値と線要素相対値にはさまれた領域に存在することを示す、即ち

解像度 $100DPI$ のプリンタに対し 10 ポイントサイズの文字を出力する場合 ($PIX_SIZE = 10 \times 100 / 72 = 14$) などがこれに相当する。

低画素時刻付けの演算

低画素時刻付け演算処理のフローチャートを第²²~~23~~図に示し、これに従って以下に説明する。

2個の演算結果用配列 $D_ZLC[0:No_of_Line]$ と $D_ZLO[0:No_of_Line]$ を $DARAM$ 上に確保する。

割付け値変数 CU_COD を初期化する ($CU_COD = 0$)。

$i = 0$ なる i について、

$D_ZLC[i] = CU_COD$

$zep[i] \leq MINPIX$ の場合、

$CU_COD = CU_COD + 1$

$D_ZLO[i] = CU_COD$

さらに、 $i = 1 \sim No_of_Line - 1$ なる i について、

$D_ZLC[i] = CU_COD$

$D_CORD[i] = (BAND[BAND_NO].$
 $ZLPO[LINE_SEQ] + Cord_off$
 $[COFF_P]) \times SCALE_F$

$COFF_P = COFF_P + 1$ 。

さらに ADG の場合、

$D_CORD[i] = ROUND(D_CORD$
 $[i])$

ZSM のとき、

線要素相対値と線要素基準値にはさまれた領域に存在することを示す、即ち

$D_CORD[i] = (BAND[BAND_NO].$
 $ZLPC[LINE_SEQ] + Cord_off$
 $[COFF_P]) \times SCALE_F$

$COFF_P = COFF_P + 1$ 。

さらに ADG の場合、

$D_CORD[i] = ROUND(D_CORD$
 $[i])$

PIX_SIZE が $MINPIX$ 以下なる場合

$zep[i] \leq MINPIX$ の場合、

$CU_COD = CU_COD + 2$

$D_ZLO[i] = CU_COD$

この結果を PIX_SIZE が 13 だった場合について具体的な例をあげると、

$D_ZLC[0:8]$ の値はそれぞれ、 $0, 0, 2, 4, 6, 6, 8, 10, 12$ に、 $D_ZLO[0:8]$ は同様に、 $0, 2, 4, 6, 6, 8, 10, 12, 12$ に割付けられる。

PIX_SIZE が 17 以下の場合の「亀」の例における水平線バンド1「亀」の部分の結果を第¹⁶~~18~~図に示す。

制御点情報配列より制御点座標値の作成

制御点情報配列より制御点座標値作成演算処理のフローチャートを第²³~~24~~図に示し、これに従って以下に説明する。

出力解像度制御点演算結果用配列 $D_CORD[0:No_of_Cp-1]$ を用意する。

REG_F の値が、

CPY のとき、

$D_CORD[i] = D_CORD[i-1]$

ZLOのとき、

$D_CORD[i] = BAND[BAND_NO]$

$D_ZLO[LINE_SEQ]$

ZLCのとき、ZLOと同様。

ZLMのとき、ZLOと同様。

ZSMのとき、

$D_CORD[i] = (BAND[BAND_NO] \cdot D_ZLO[LINE_SEQ] + BAND[BAND_NO] \cdot D_ZLC[LINE_SEQ]) / 2$

ADGはこの処理で必要としない。

PIX_SIZEがその他(例えばMINPIXより大かつMINPIX×5より小)の場合

解像度400DPIのプリンタに対し12ポイントサイズの文字を出力する場合($PIX_SIZE = 12 \times 400 / 72 = 67$)などがこれに相当する。

A < Bなら、

$D_ZLC[i] = D_ZLC[i] - 1$

上記調整演算はその簡略化として以下のような別法を用いてもよい。

一法として、

$D_ZLO[i] = D_ZLO[i] + 1$

あるいは、

$D_ZLC[i] = D_ZLC[i] - 1$

ドット展開

以上のような処理で作成された出力解像度合致制御点は、次にドット展開回路に入力される。

ドット展開回路は前述の如く単純なビットマップ塗り潰し回路が望ましく、ビットマップの領域を初期化(全てのビットをOFFする)した後、先ず出力解像度合致制御点に沿った輪郭線が発生(該当するビットをONする)し、その後ビットマップ上のONビットからその次のONビットまでを全てONすることによりドット展開が実行される。

根要素基準値の演算(演算結果用配列D_ZLO[0:No_of_Line-1]の作成)および線要素相対値の演算(演算結果用配列D_ZLC[0:No_of_Line-1]の作成)に関してはPIX_SIZEがMINPIXに対して十分大なる場合と同等におこない、その直後に以下のスペース割付け調整処理を追加する。

スペース割付け調整処理

スペース割付け調整処理演算処理のフローチャートを第20図に示し、これに従って以下に説明する。

$i = 1 \sim No_of_Line - 1$ なるiに対し、 $SP_B = D_ZLO[i] - D_ZLC[i]$ の値をチェックし、 $SP_B < 2$ の場合、以下の演算による調整をおこなう。

$A = D_ZLC[i+1] - D_ZLO[i]$

$B = D_ZLC[i] - D_ZLO[i-1]$

なるAとBを比較し、

$A \geq B$ なら、

$D_ZLO[i] = D_ZLO[i] + 1$

出力機器への転送

ドット展開されたデータは前述入力パラメータにて指定された出力機器に出力するため、その出力機器に連結される第1図21に示すフレームバッファのいずれかに転送される。

[発明の効果]

以上説明したように、標準的な図形情報として用意された閉輪郭座標情報と、これに含まれる複数の閉輪郭を区別するための輪郭情報を、指定された出力サイズに応じて拡大または縮小演算した後、ドット情報に変換し、出力する文字処理装置が用いるビットマップ展開用文字データに変換するするに際し、

外輪郭、内輪郭を判別する手段と、

水平線、垂直線を抽出する手段と、

これを分類した後、水平線、垂直線の要素組を決定する手段と、

分類する手段と、

水平、垂直スペース領域を設定する手段と、

低画素時省略順位を設定する手段とを具備し、

当該文字の輪郭を表現するための制御点座標と、

複数の閉輪郭を区別する輪郭情報と、

バンド情報と、

前述バンド情報を参照するための制御点情報と、

前述バンド情報に含まれない制御点の座標値情報

からなるビットマップ展開用文字データに変換することにより、

広範囲のサイズで、高品位の文字を、出力可能なビットマップ展開用文字データを効率良く得ることが可能となった。

4. 図面の簡単な説明

第1図は、本発明のシステム構成図

第2図は、フォント座標系を示す図

第3図は、標準となる文字データの制御点配置を示す図

第4図は、文字輪郭制御点データを示す図

第5図は、標準となる文字データよりビットマッ

プ変換用文字データを自動的に作成する処理のフローチャート

第6図は、外輪郭、内輪郭判別処理のフローチャート

第7図は、第6図フローチャートにおける処理を補足説明する図

第8図は、バンド情報を作成する処理のフローチャート

第9図は、「種」の水平線要素組および垂直線要素組の決定結果の例を示す図

第10図は、第9図に対応した文字図形における反映例

第11図は、低画素時省略順位設定処理のフローチャート

第12図は、「種」のバンドデータを示す図

^{1.8}
~~第4図~~ 第13図は、本システムにおける文字処理装置部分の構成図

^{1.4}
~~第4図~~ 第14図は、ビットマップ変換用文字データの格納形式を示す図

^{1.5}
~~第4図~~ 第15図は、制御点情報を示す図

^{1.6}
~~第4図~~ 第16図は、PIX_SIZEが17以下の場合のピクセル割付け方法を示す図

^{1.7}
~~第4図~~ 第17図は、ビットマップ文字の出力を説明するフローチャート

^{1.8}
~~第4図~~ 第18図は、線要素基準値の演算処理のフローチャート

^{1.9}
~~第4図~~ 第19図は、線要素相対値の演算処理のフローチャート

^{2.0}
~~第4図~~ 第20図は、スペース割付け調整演算処理のフローチャート

^{2.1}
~~第4図~~ 第21図は、制御点情報配列より制御点座標値作成演算処理のフローチャート

^{2.2}
~~第4図~~ 第22図は、低画素時割付け演算処理のフローチャート

^{2.3}
~~第4図~~ 第23図は、制御点情報配列より制御点座標値作成演算処理のフローチャート

5は、ドキュメントファイル格納領域、

6は、キーボード装置、

7および8は、プログラム用RAM、

9は、データ用RAM、

10、11、12、および13は、フレームバッファ、

18は、CPU、

19は、ディスク装置、

22は、システムの機器構成などを記憶してあるROM

出願人 キヤノン株式会社

代理人 丸 島 儀 一

西 山 恵 三

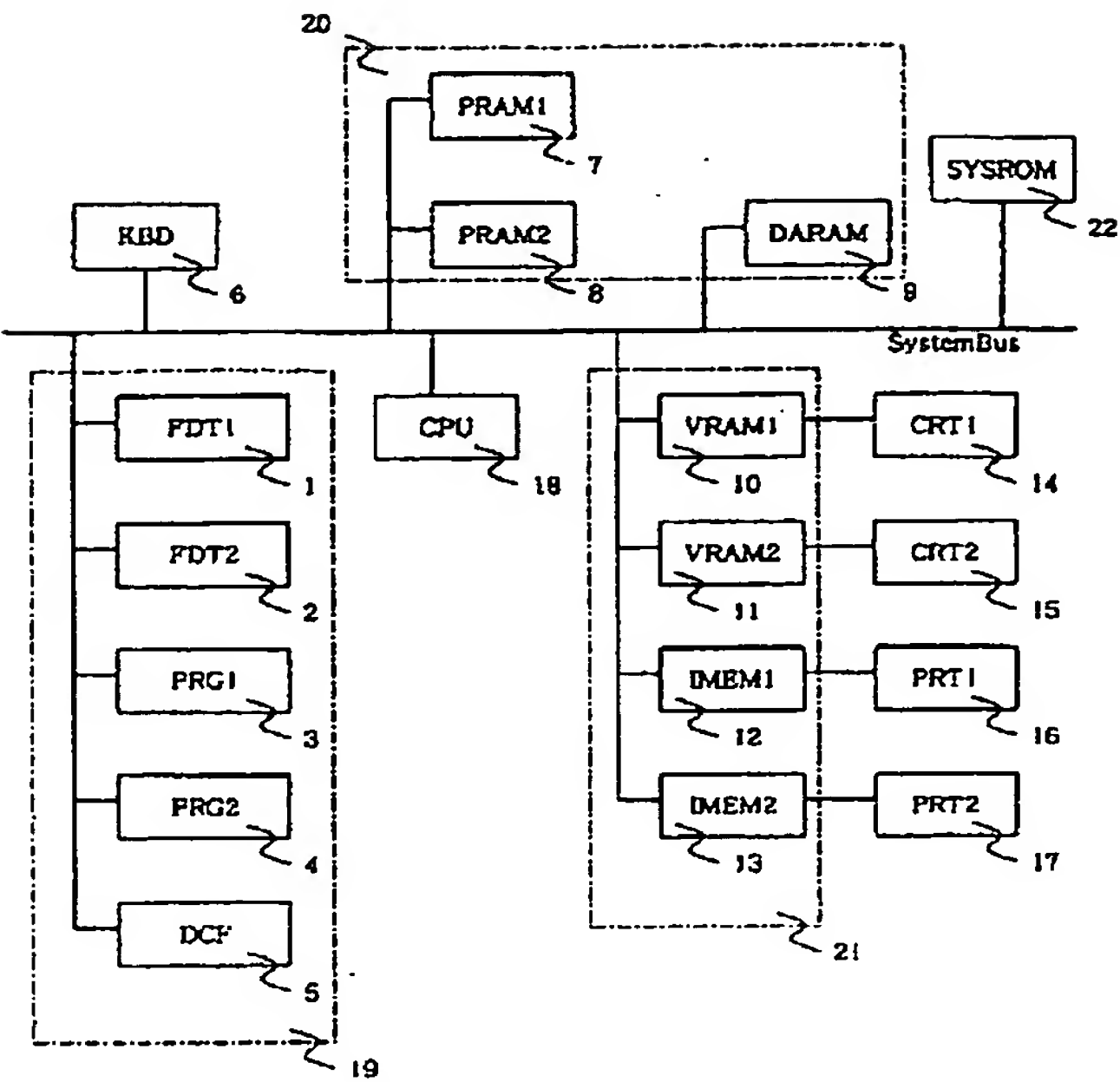


1は、標準となる文字データ格納領域、

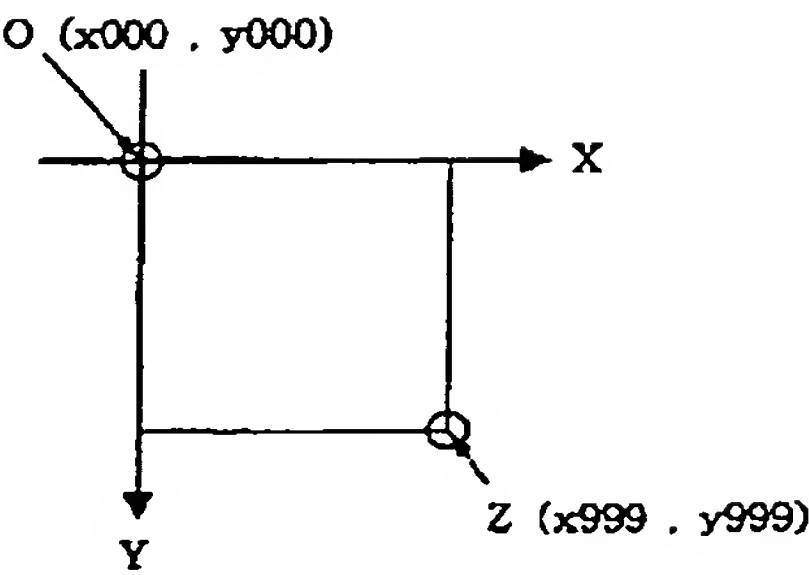
2は、ビットマップ変換用文字データ格納領域、

3および4は、プログラム格納領域、

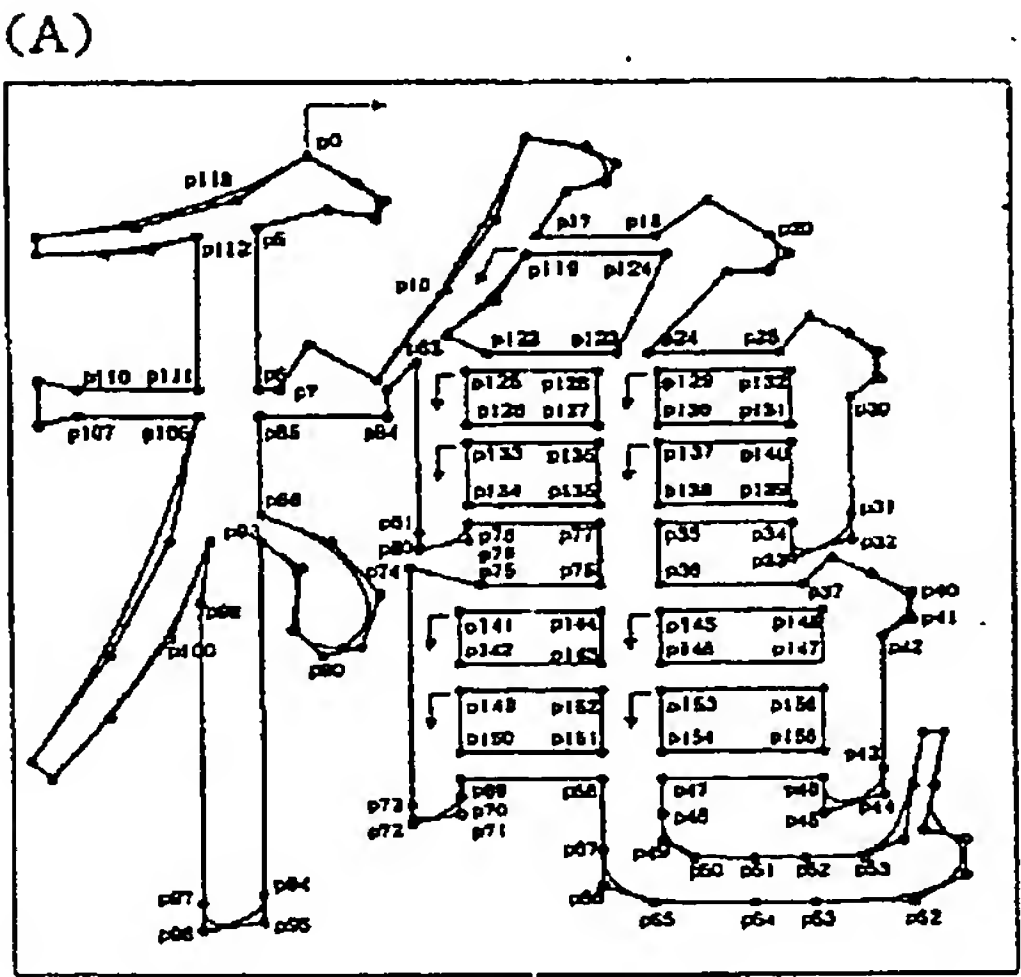
第1図



第2図

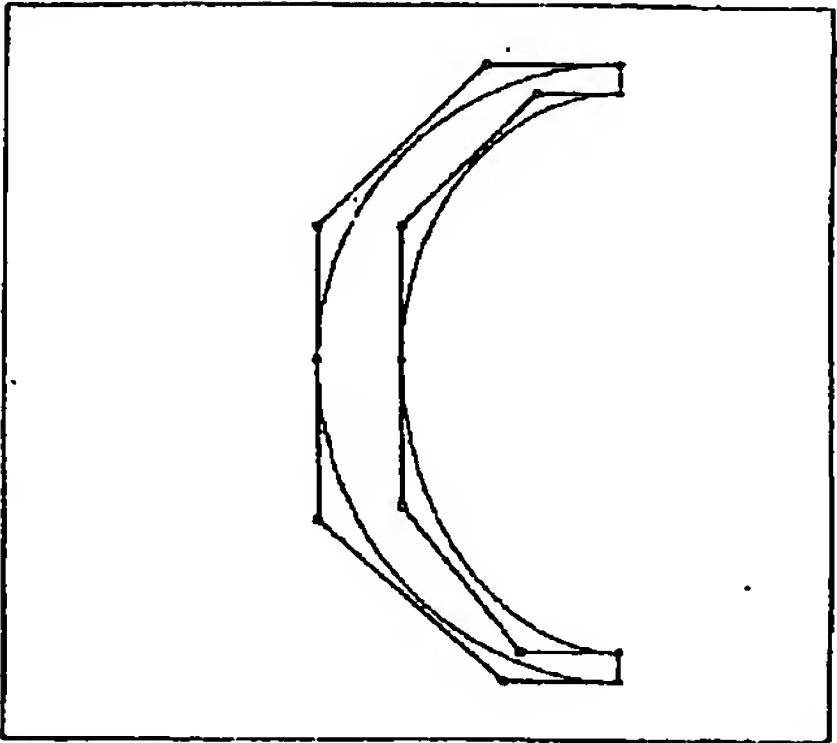


第3図

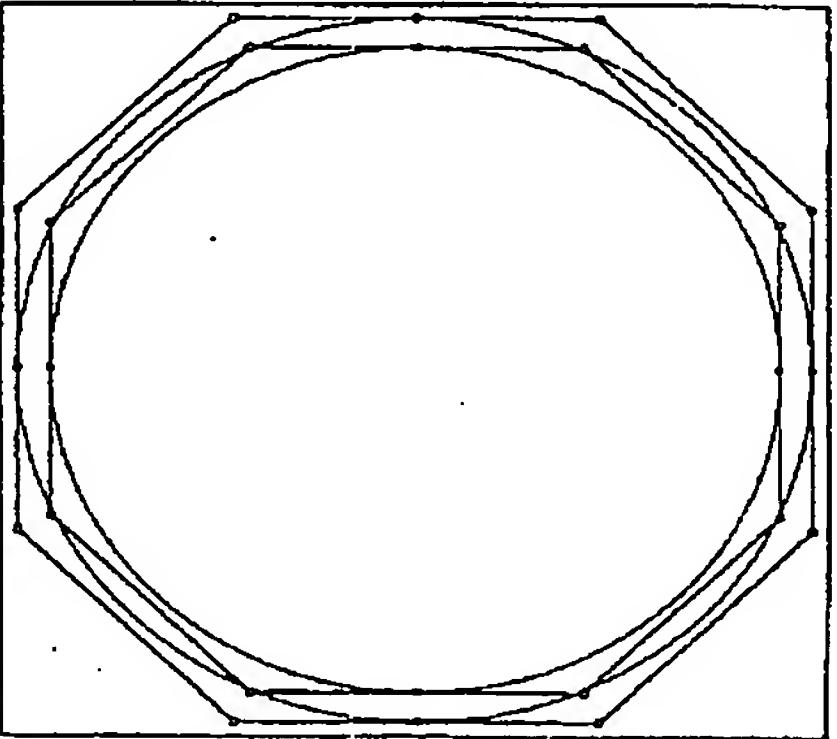


第3図

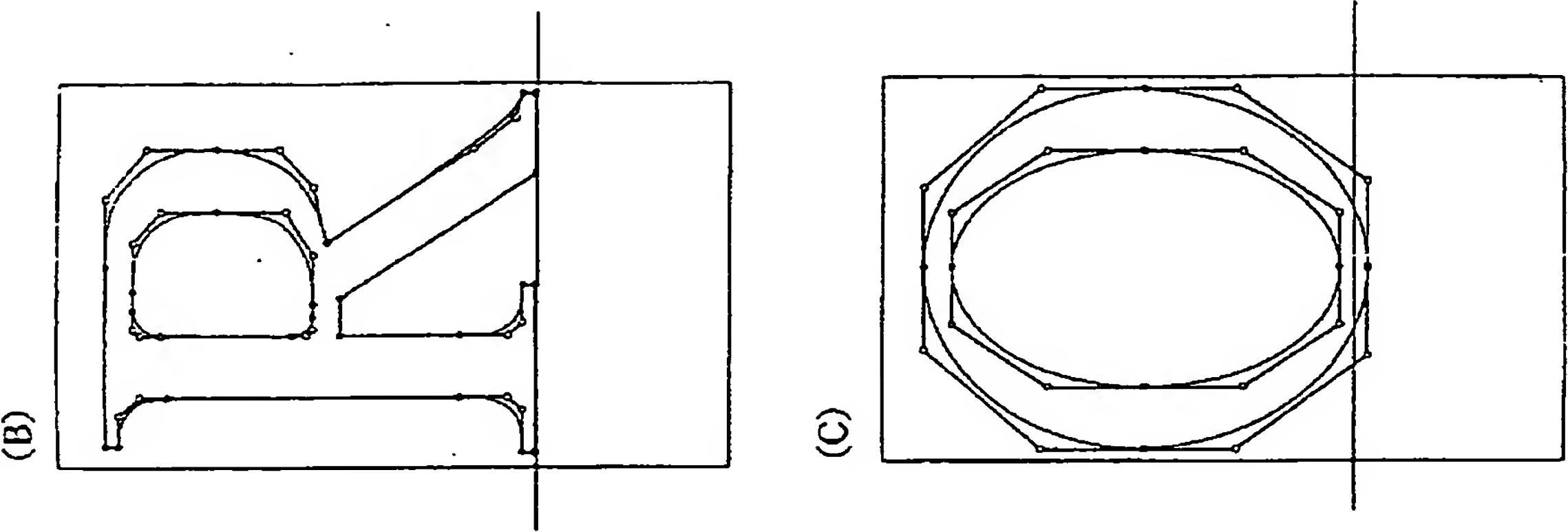
(D)



(E)



第3図



第4図

(A)

輪郭制御点配列			
P	X	Y	C
p0	x0	y0	c0
p1	x1	y1	c1
p2	x2	y2	c2
p3	x3	y3	c3
p4	x4	y4	c4
p5	x5	y5	c5
p6	x6	y6	c6
p7	x7	y7	c7
p8	x8	y8	c8
p9	x9	y9	c9
p10	x10	y10	c10
p11	x11	y11	c11
p12	x12	y12	c12
p13	x13	y13	c13
p14	x14	y14	c14
p15	x15	y15	c15
p16	x16	y16	c16
p17	x17	y17	c17
p18	x18	y18	c18
p19	x19	y19	c19
p20	x20	y20	c20
p21	x21	y21	c21
p22	x22	y22	c22
p23	x23	y23	c23
p24	x24	y24	c24
p25	x25	y25	c25
p26	x26	y26	c26
p27	x27	y27	c27
p28	x28	y28	c28
p29	x29	y29	c29

t0⇒

p100	x100	y100	c100
p101	x101	y101	c101
p102	x102	y102	c102
p103	x103	y103	c103
p104	x104	y104	c104
p105	x105	y105	c105
p106	x106	y106	c106
p107	x107	y107	c107
p108	x108	y108	c108
p109	x109	y109	c109
p110	x110	y110	c110
p111	x111	y111	c111
p112	x112	y112	c112
p113	x113	y113	c113
p114	x114	y114	c114
p115	x115	y115	c115
p116	x116	y116	c116
p117	x117	y117	c117
p118	x118	y118	c118
p119	x119	y119	c119
p120	x120	y120	c120
p121	x121	y121	c121
p122	x122	y122	c122
p123	x123	y123	c123
p124	x124	y124	c124
p125	x125	y125	c125
p126	x126	y126	c126
p127	x127	y127	c127
p128	x128	y128	c128
p129	x129	y129	c129

t1⇒

t2⇒

↓

↓

↓

8

↓

p130	x130	y130	c130
p131	x131	y131	c131
p132	x132	y132	c132
p133	x133	y133	c133
p134	x134	y134	c134
p135	x135	y135	c135
p136	x136	y136	c136
p137	x137	y137	c137
p138	x138	y138	c138
p139	x139	y139	c139
p140	x140	y140	c140
p141	x141	y141	c141
p142	x142	y142	c142
p143	x143	y143	c143
p144	x144	y144	c144
p145	x145	y145	c145
p146	x146	y146	c146
p147	x147	y147	c147
p148	x148	y148	c148
p149	x149	y149	c149
p150	x150	y150	c150
p151	x151	y151	c151
p152	x152	y152	c152
p153	x153	y153	c153
p154	x154	y154	c154
p155	x155	y155	c155
p156	x156	y156	c156

t4⇒

t5⇒

t6⇒

t7⇒

t8⇒

t9⇒

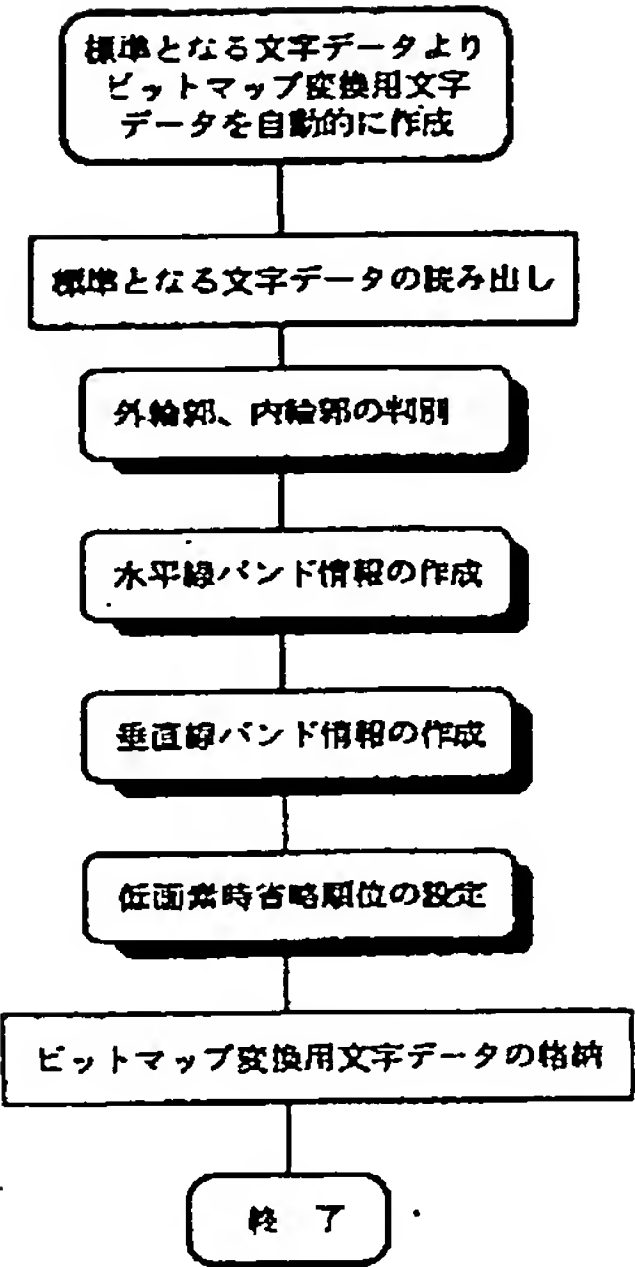
第5図

第4図

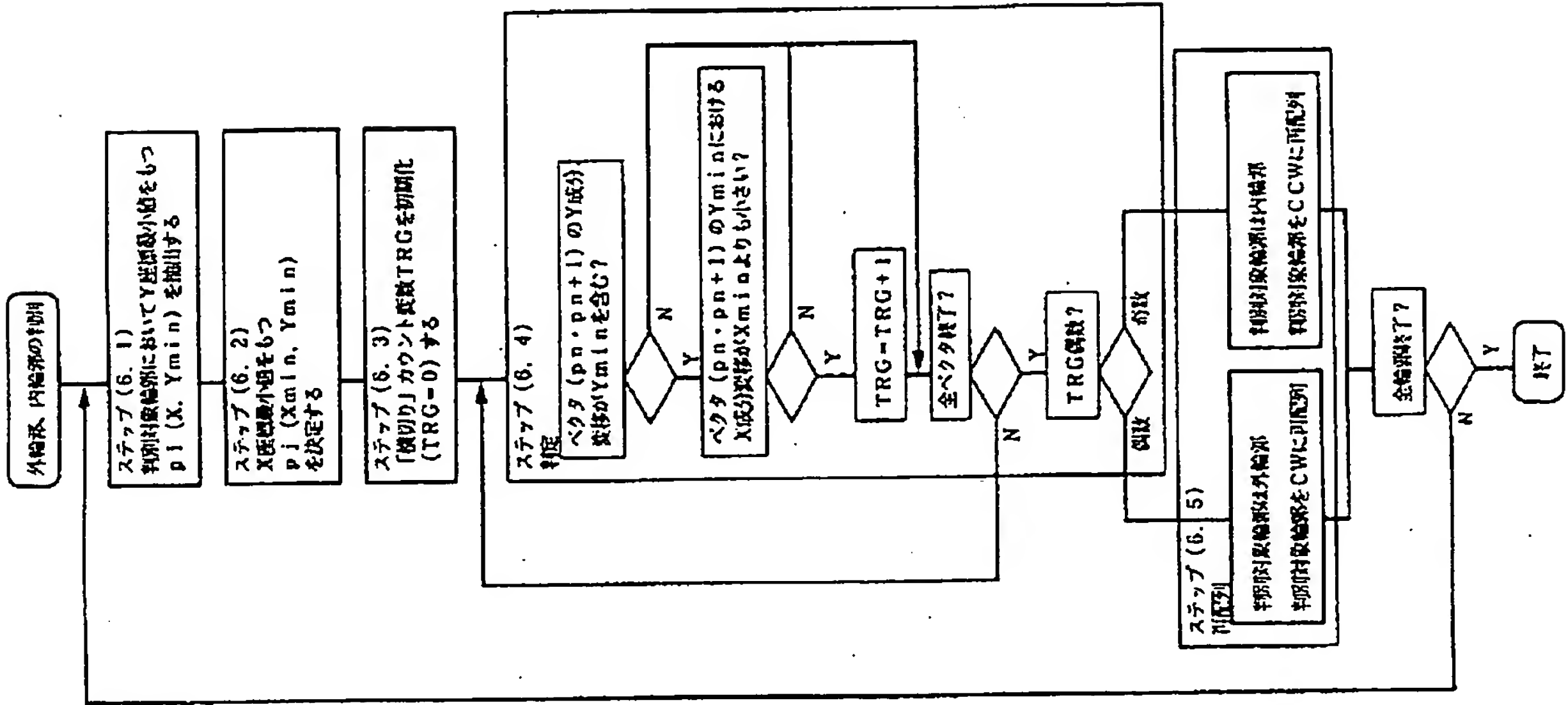
(B)

輪郭情報		
制御点総数 No_of_Cp		
開始点総数 nt		
t0	p0	p118
t1	p119	p124
t2	p125	p128
t3	p129	p132
t4	p133	p136
t5	p137	p140
t6	p141	p144
t7	p145	p148
t8	p149	p152
t9	p153	p156

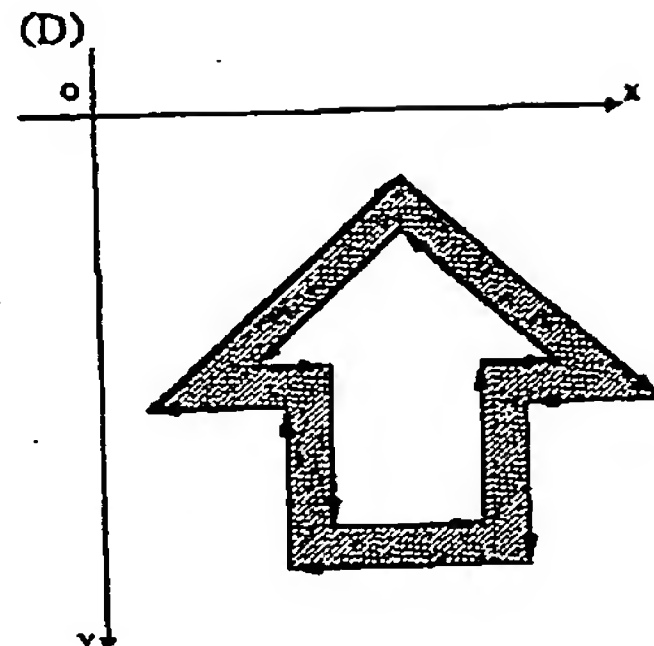
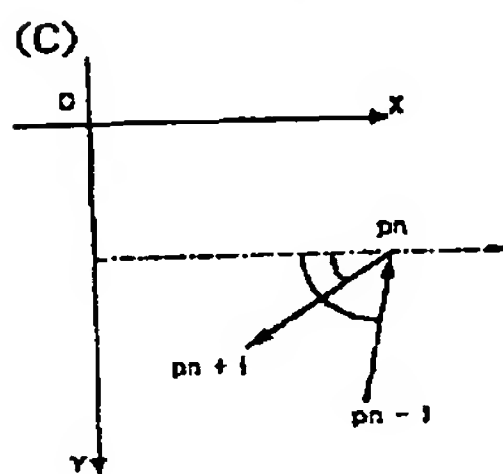
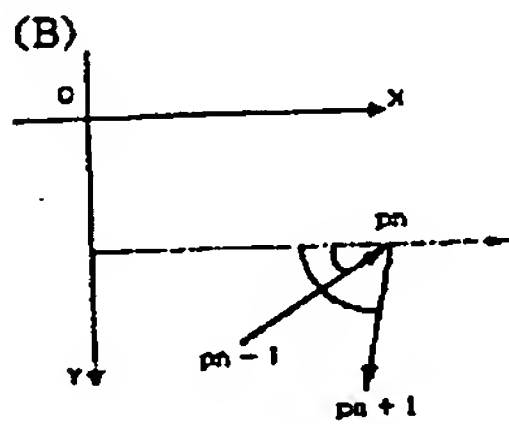
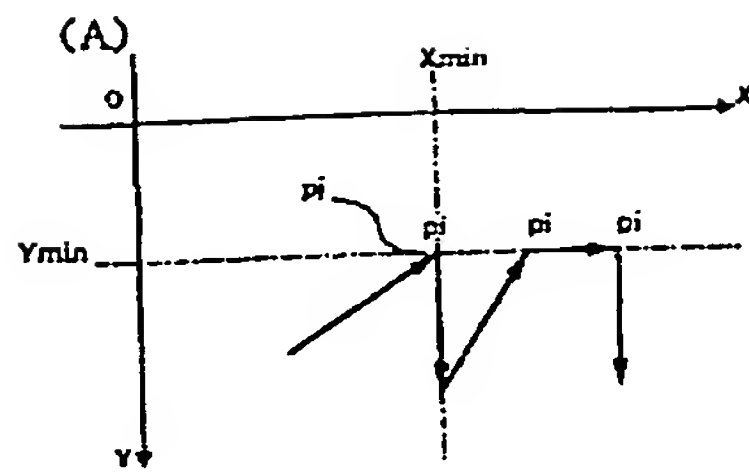
字形情報		
min_p	x103	y12
max_p	x60	y98



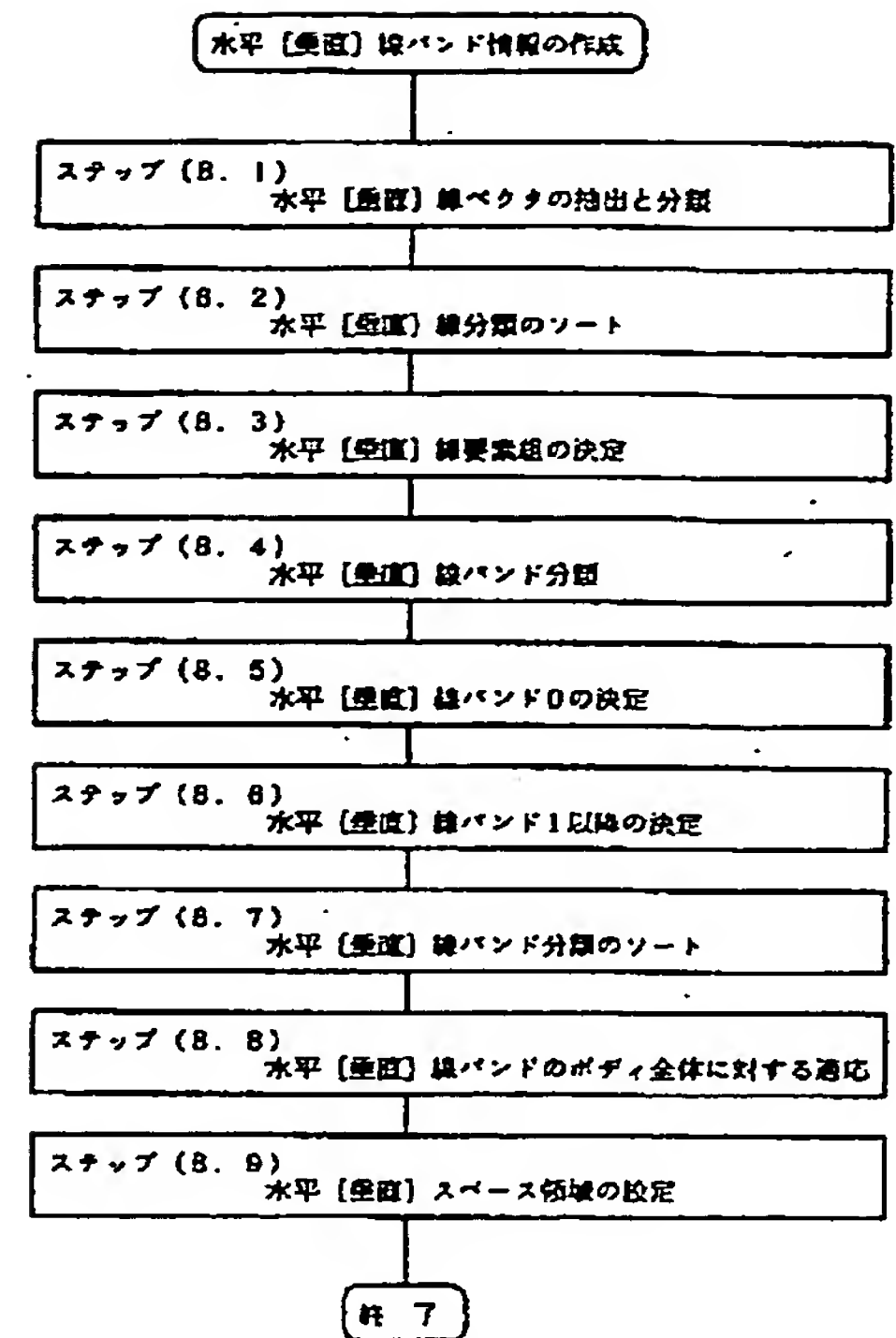
第6図



第7図



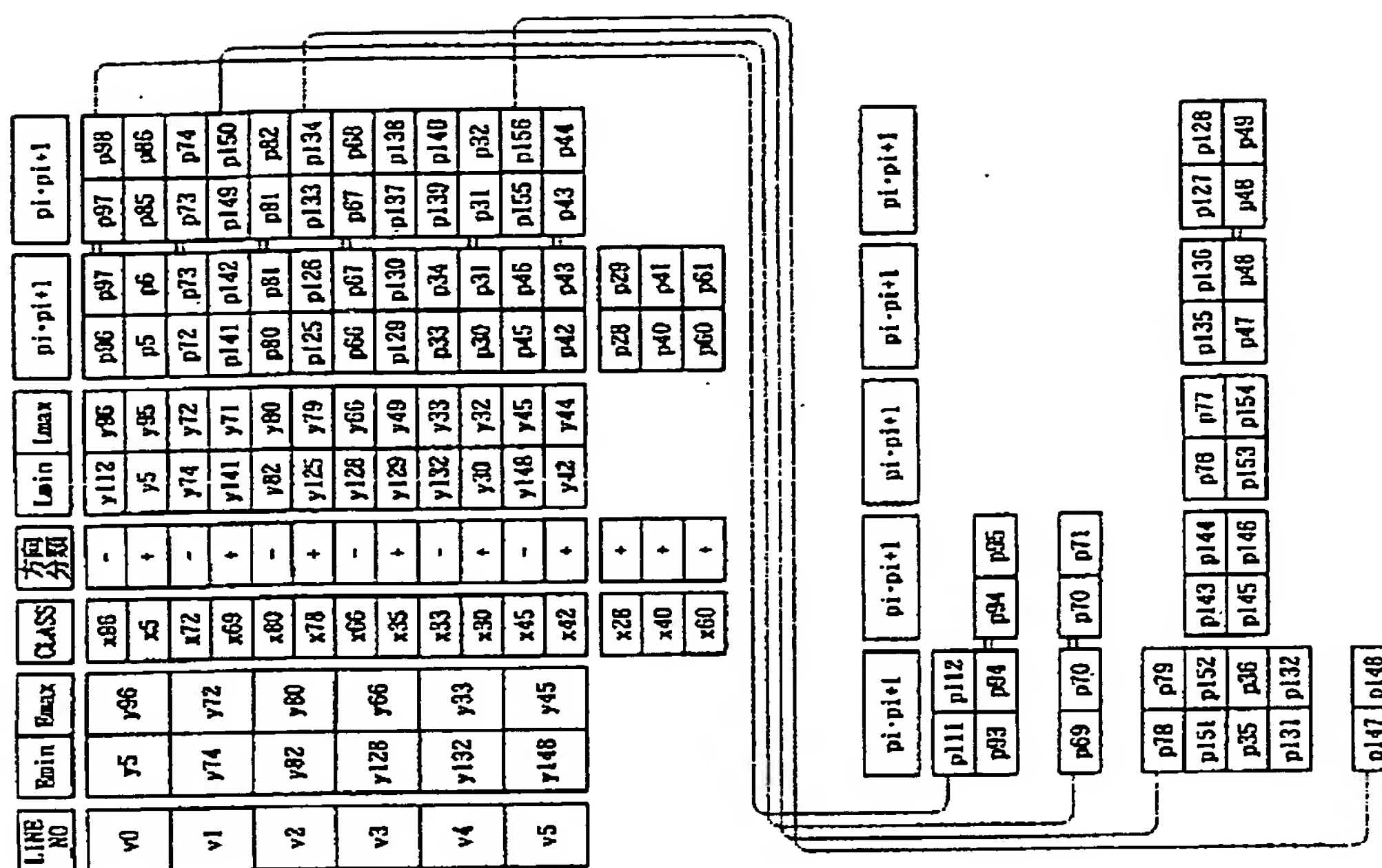
第8図



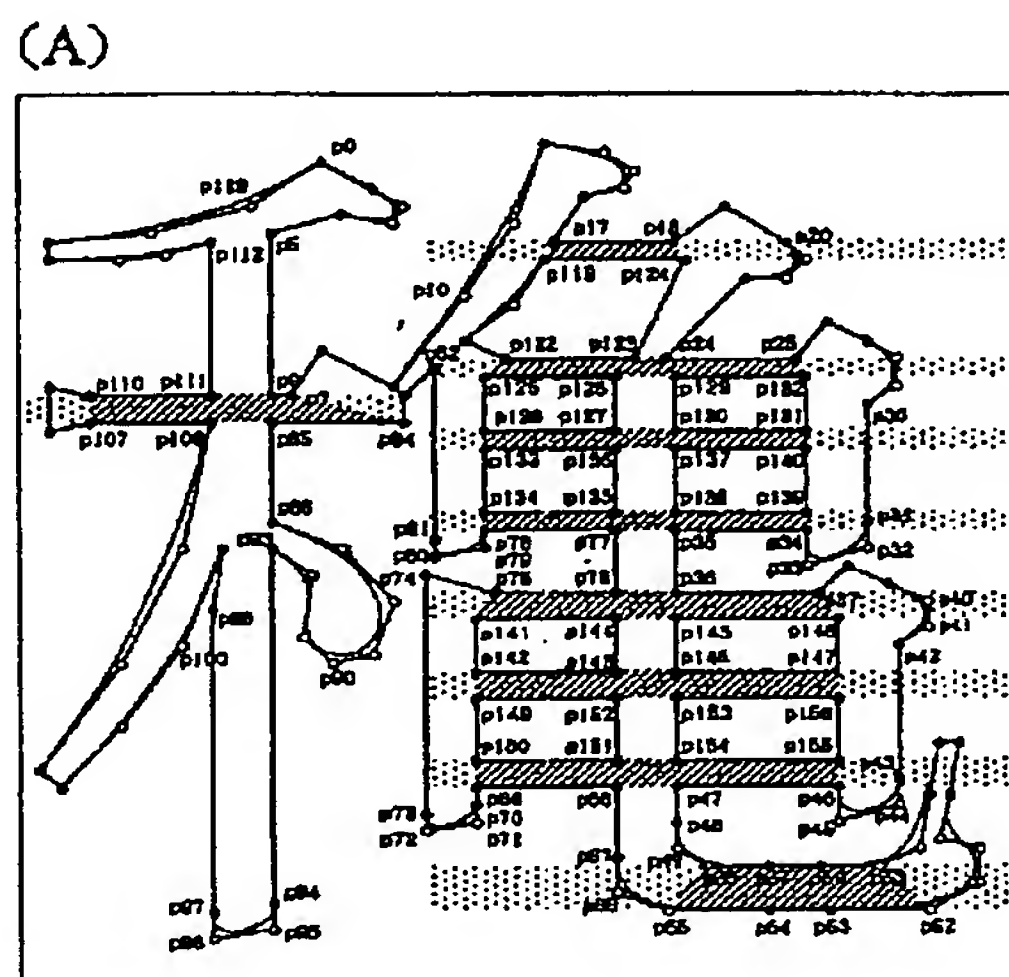
第9図 (A)

LINE NO	Emin	Emax	CLASS	方向分類	Lmin	Lmax	p1-p1+1	p1-p1+1	p1-p1+1
h0	x119	x124	y17	+	x17	x18	p17	p18	
			y119	-	x119	x124	p124	p119	
h1	x125	x132	y24	+	x122	x25	p122	p123	p24
			y125	-	x125	x132	p132	p129	p128
h2	x107	x84	y8	+	x110	x7	p110	p111	p6
			y84	-	x107	x84	p84	p85	p106
h3	x126	x131	y126	+	x126	x131	p126	p127	p130
			y133	-	x133	x140	p140	p137	p136
h4	x78	x34	y134	+	x134	x138	p134	p135	p138
			y24	-	x78	x34	p34	p35	p77
h5	x141	x148	y36	+	x75	x97	p75	p76	p36
			y141	-	x141	x148	p148	p145	p144
h6	x142	x147	y142	+	x142	x147	p142	p143	p146
			y149	-	x149	x156	p156	p153	p152
h7	x89	x46	y150	+	x150	x155	p150	p161	p164
			y48	-	x89	x46	p48	p47	p68
h8	x65	x62	y50	+	x50	x53	p50	p51	p52
			y62	-	x65	x62	p62	p63	p64
			y113	-			p113	p114	
			y22	-			p22	p23	

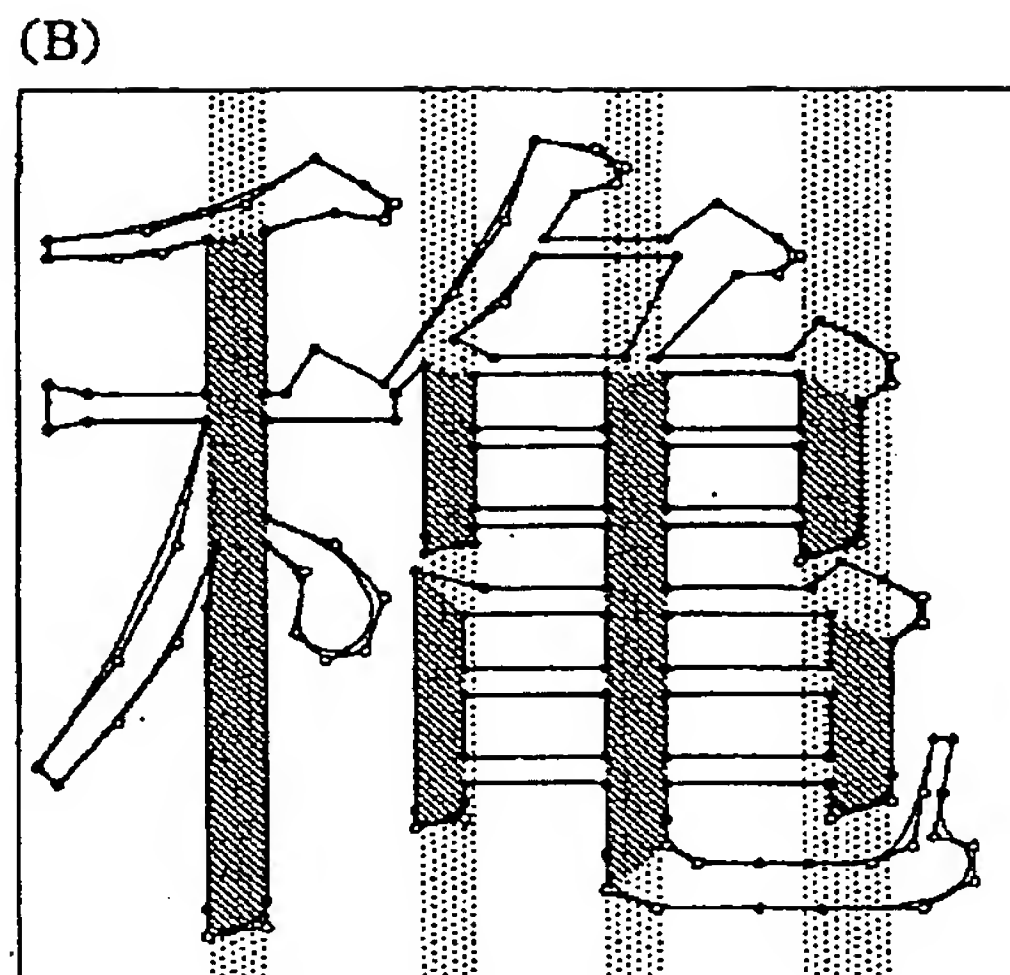
第9図 (B)



第10図

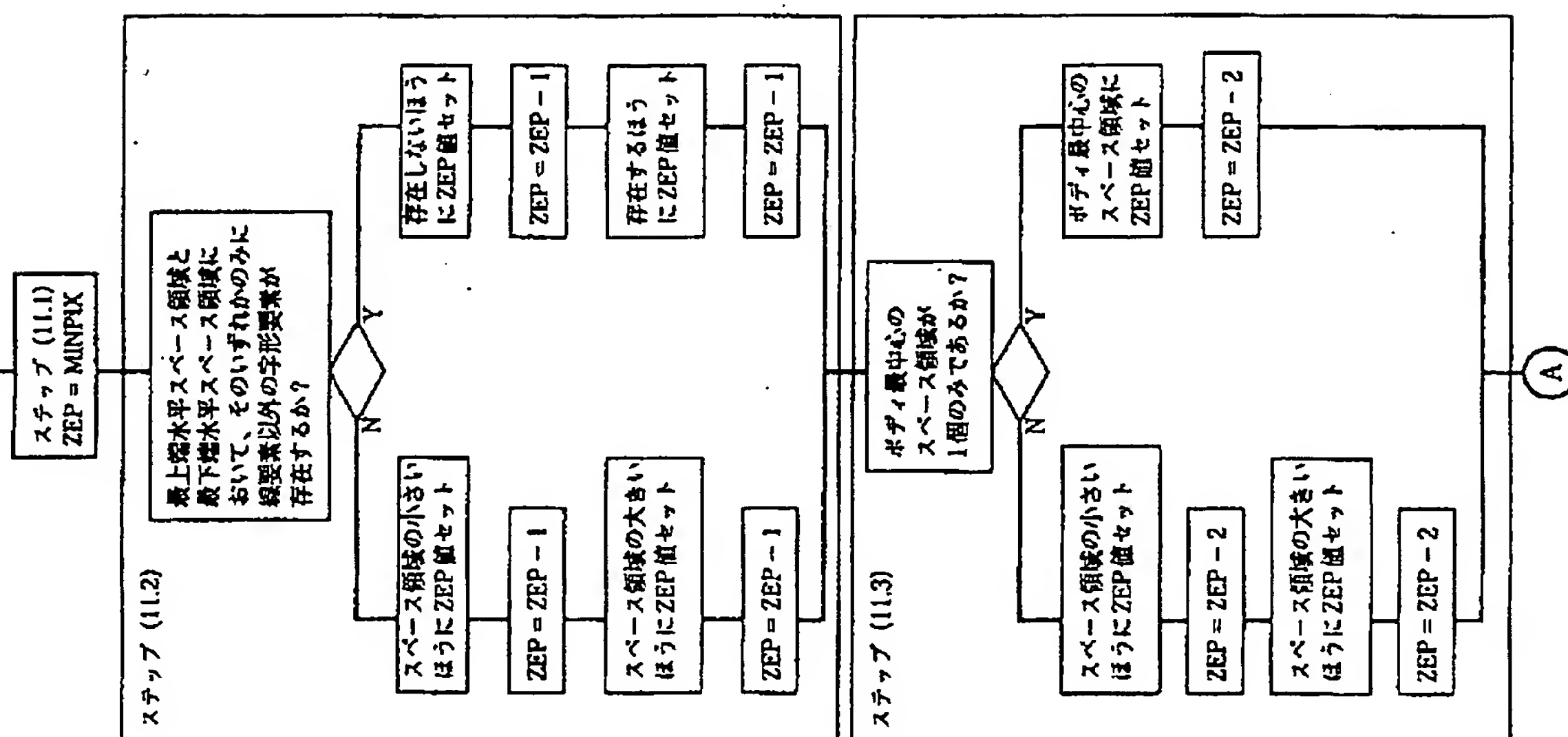


第10図

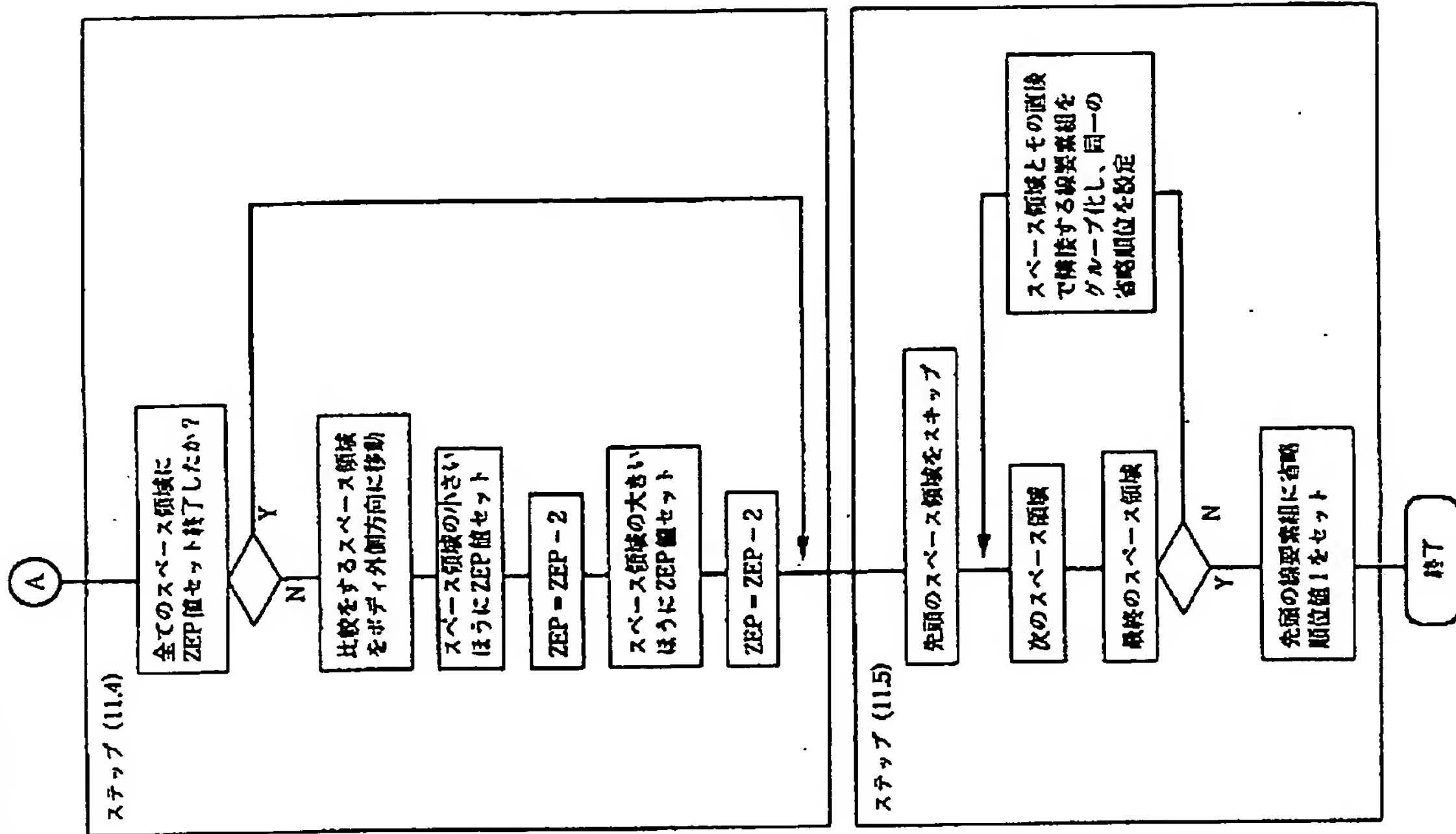


低図素時省略順位の設定

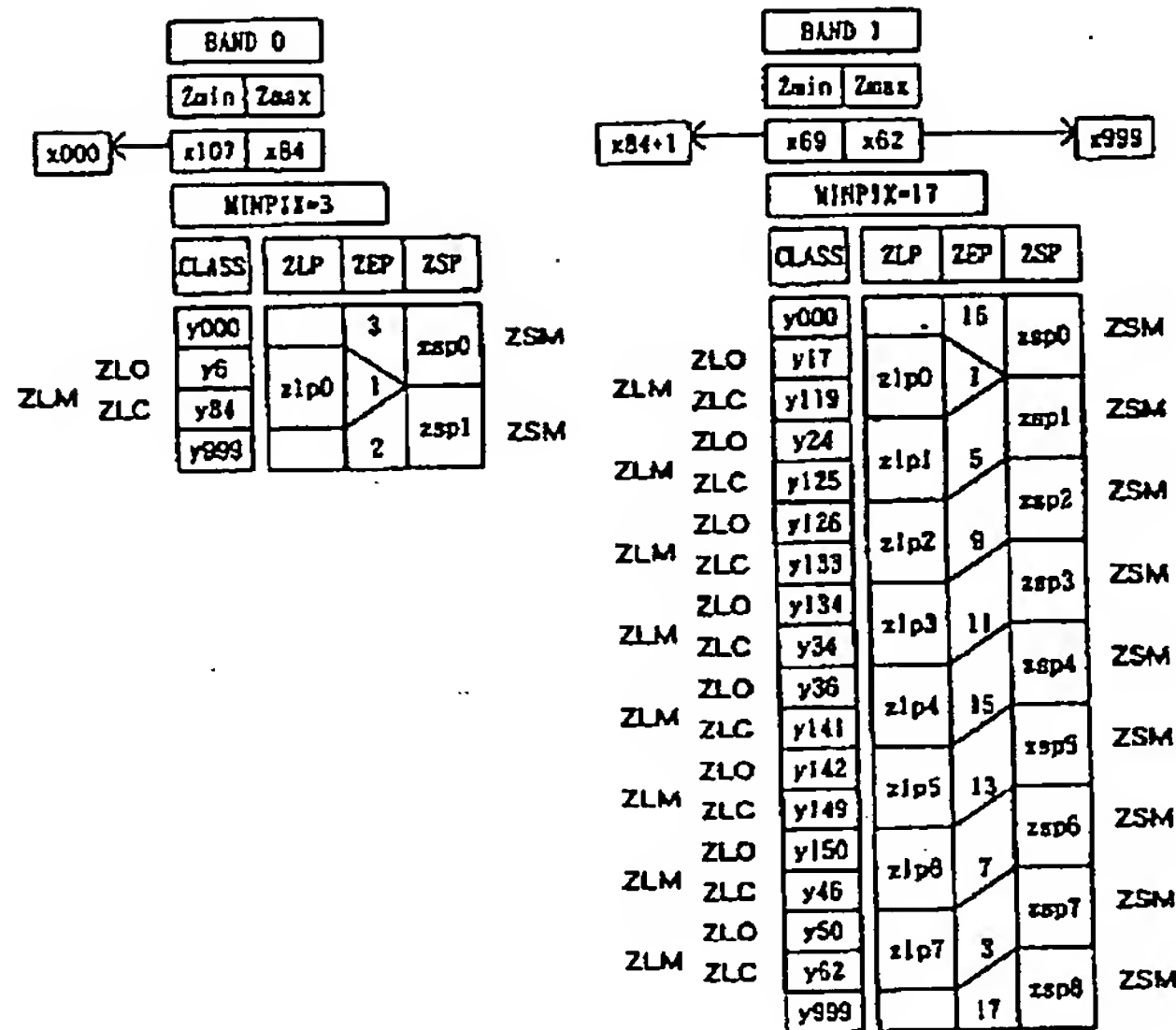
第11図 (A)



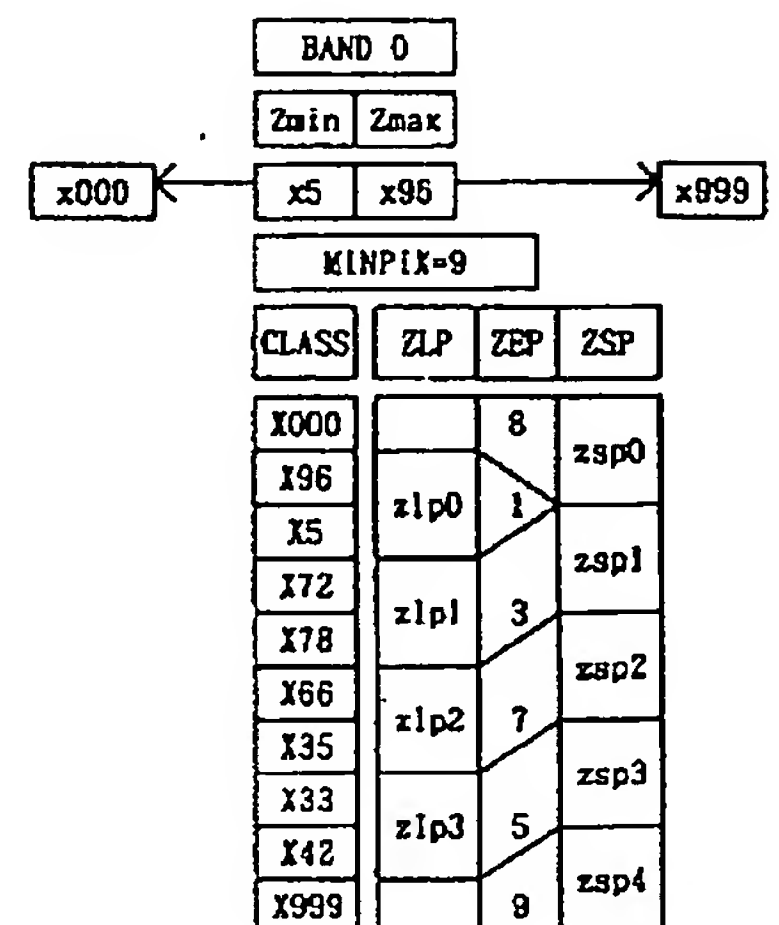
第11図 (B)



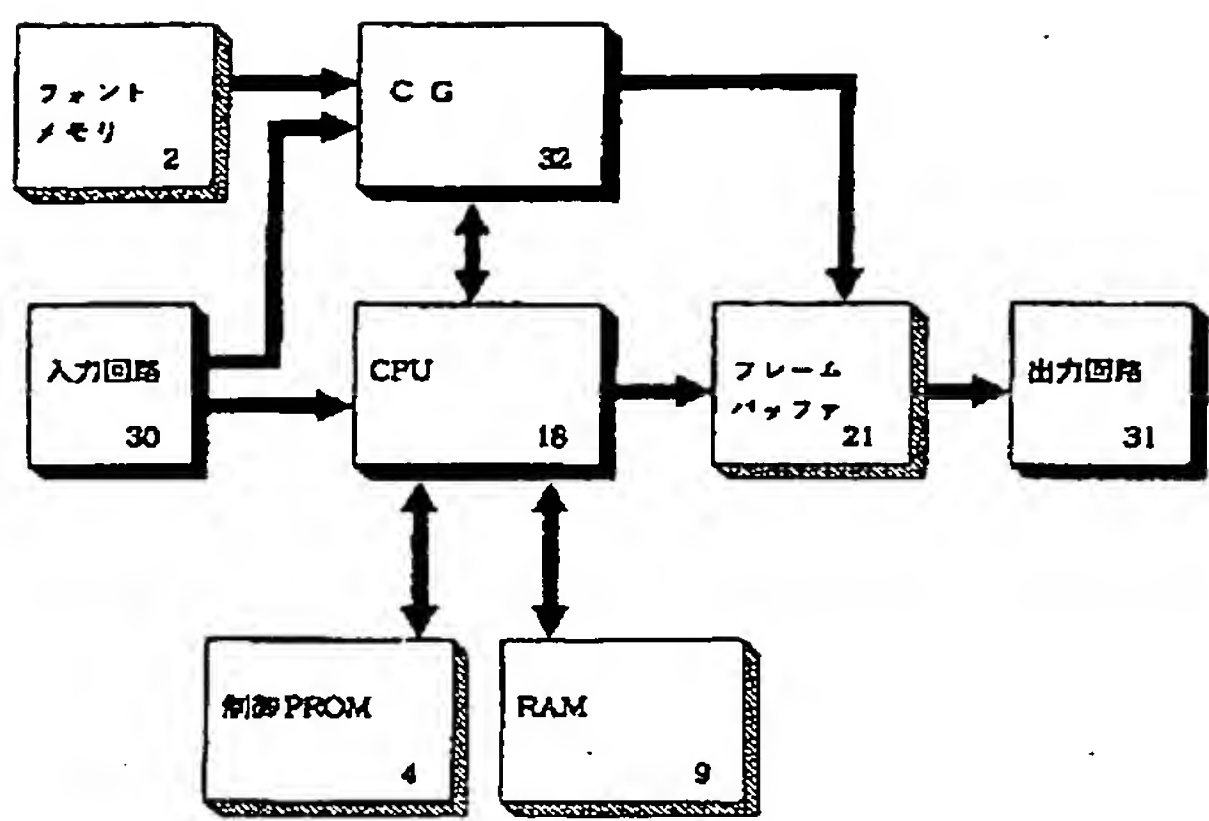
第12図 (A)



第12図 (B)



第13図



第14図 (A)

CONT_INF	No_of_Cont
	Cep[0]
	Cep[1]
	Cep[No_of_Cont-1]
OFFSET_INF	X_BAND_INF_OFF
	X_CORD_INF_OFF
	Y_BAND_INF_OFF
	Y_CORD_INF_OFF
X_INF	X_BAND_INF
	X_CORD_INF
Y_INF	Y_BAND_INF
	Y_CORD_INF

第14図 (B)

BAND_INF	BAND	MINPIX	
		No_of_Band	
		Band_Offset	zoff[0]
			zoff[1]
			0
			zoff[No_of_Band-1]
	BAND[0]	No_of_Line	
		ZEP_Array	zep[0]
			zep[1]
			0
			zep[No_of_Line]
		ZLPO_Array	zipo[0]
			zipo[1]
			0
			zipo[No_of_Line]
		ZLPC_Array	zipc[0]
			zipc[1]
			0
			zipc[No_of_Line×2]
	BAND[1]	0	
	0	0	
	BAND [No_of_Band-1]	0	

第14図 (C)

CORD_INF	Inf[0]	BAND #
		CLASS_SIBQ
		REG_F
	Inf[1]	0
	0	0
	Inf [No_of_Cp-1]	0
		Cord_off[0]
		Cord_off[1]
	CORD_OFFSET	0
		Cord_off[n]

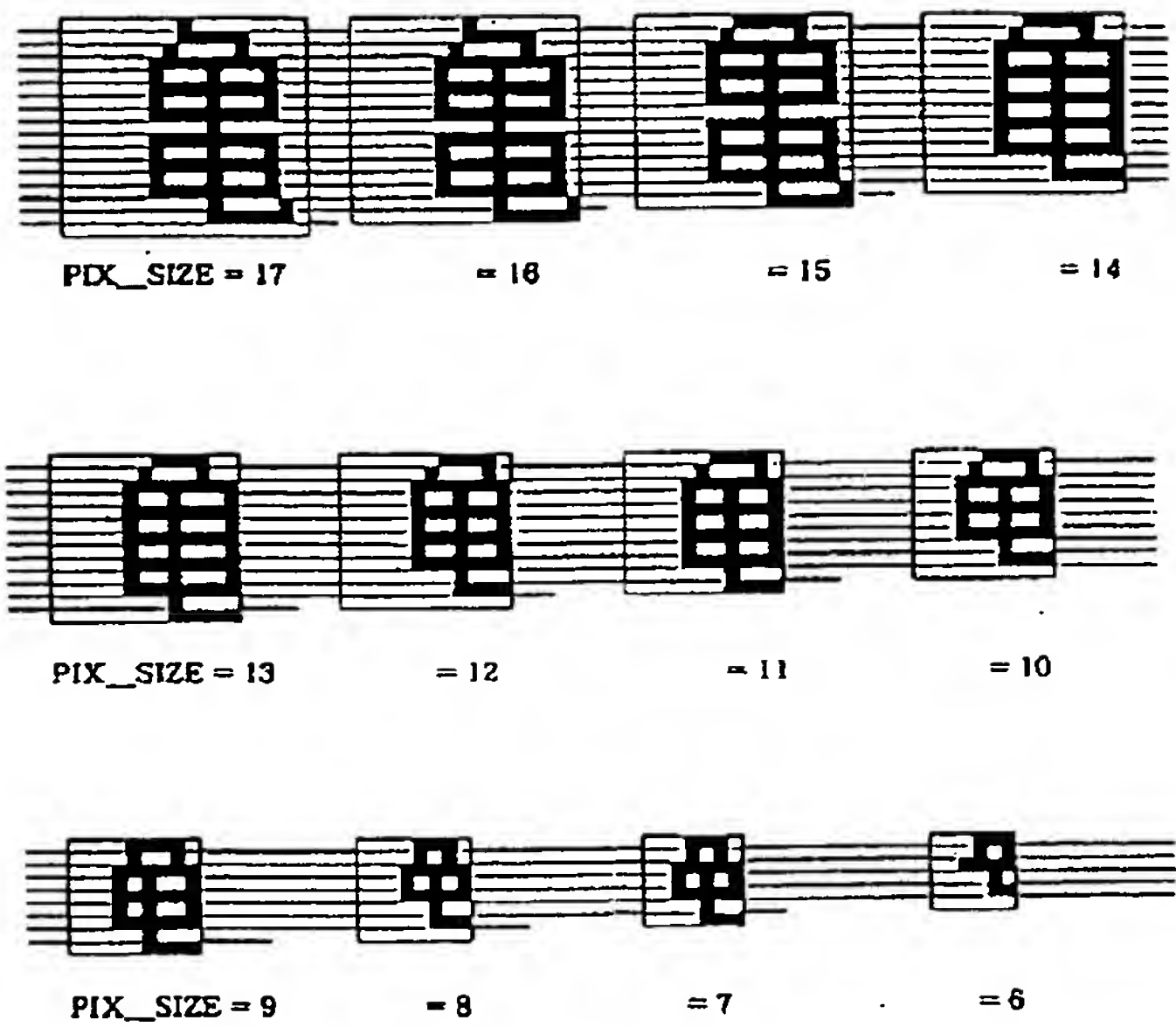
第15図

	RAND_NO	LINE_SEQ	REG_F (00000000)			
			CPY	CLS	ADG	CYC
[0]	0	0		ZSM		
[5]	0	0		ZSM		
[6]	0	0		ZLO		
[7]	0	0	CPY	ZLO		
[8]	0	0		ZSM		
[9]	0	0		ZSM		CS
[10]	1	1		ZSM		CM
[19]	1	0		ZSM		
[20]	1	0		ZLM		CS
[21]	1	0		ZLM		CM
[22]	1	1		ZSM	ADG	CM
[23]	1	1	CPY	ZSM	ADG	CE
[24]	1	2		ZLO		
[25]	1	2	CPY	ZLO		
[26]	1	1		ZSM		
[27]	1	1		ZSM		CS
[28]	1	1		ZLM		CM
[29]	1	2		ZSM		CM
[30]	1	2		ZSM		CS
[31]	1	3		ZLM		CS
[32]	1	4		ZSM		CM
[33]	1	4		ZSM		CM
[34]	1	3		ZLC		CS
[35]	1	3	CPY	ZLC		
[36]	1	4		ZLO		
[37]	1	4	CPY	ZLO		
[38]	1	4		ZSM		
[39]	1	4		ZSM		CS
[40]	1	4		ZLM		CM
[156]	1	6		ZLO		

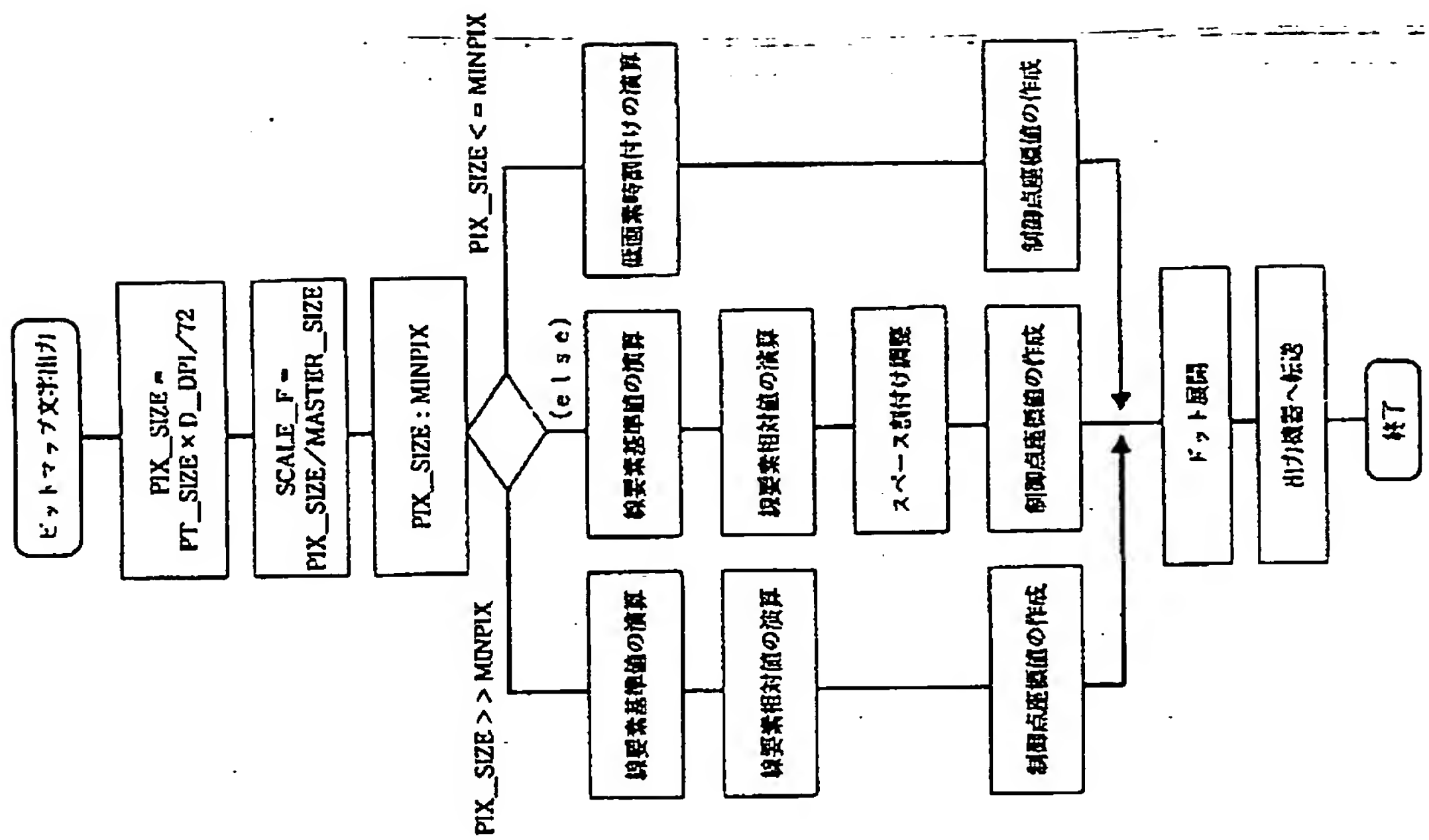
Inf_Array

ZEP_Array	
[0]	18
[1]	5
[2]	9
[3]	11
[4]	15
[5]	13
[6]	7
[7]	3
[8]	17

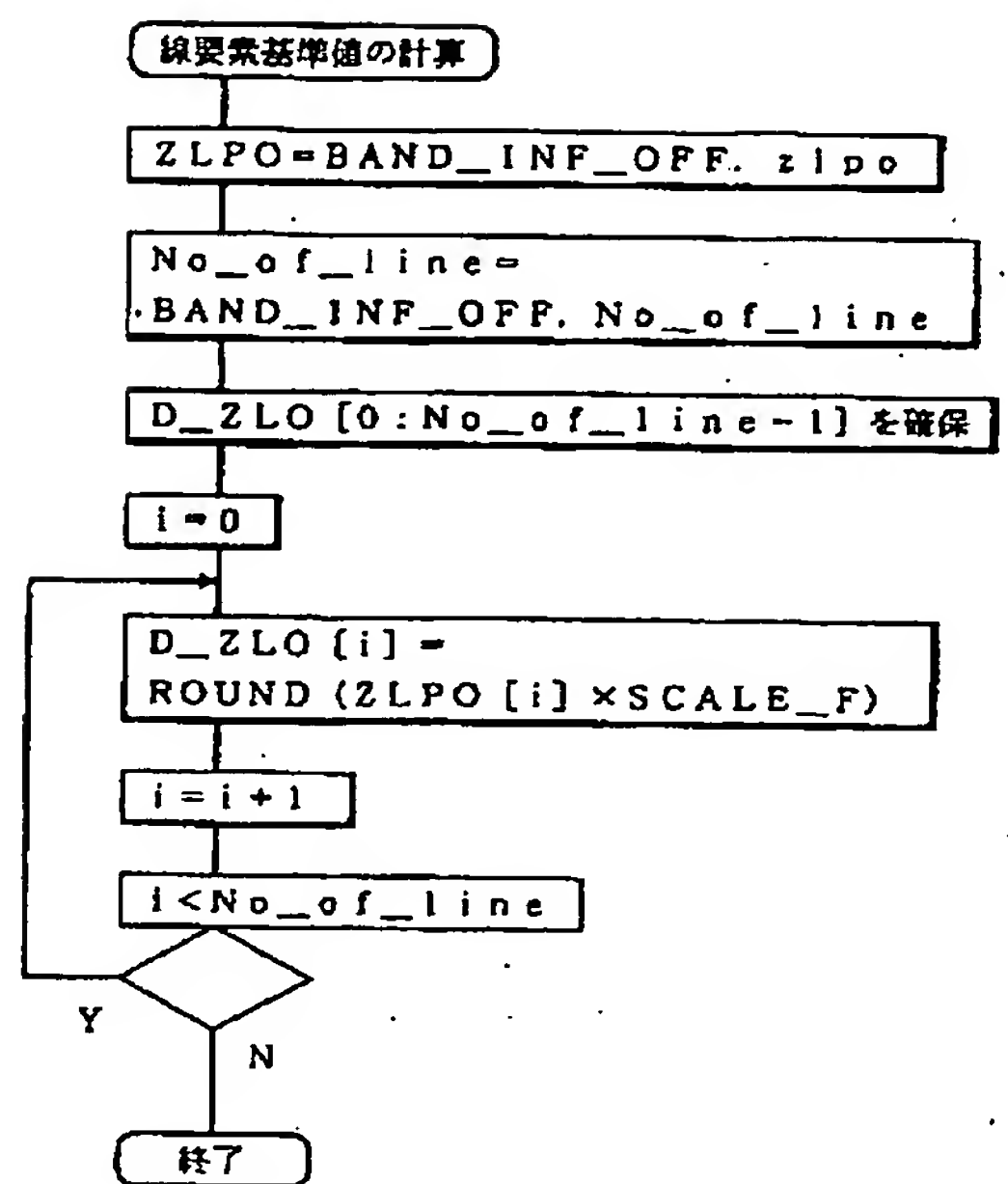
第16図



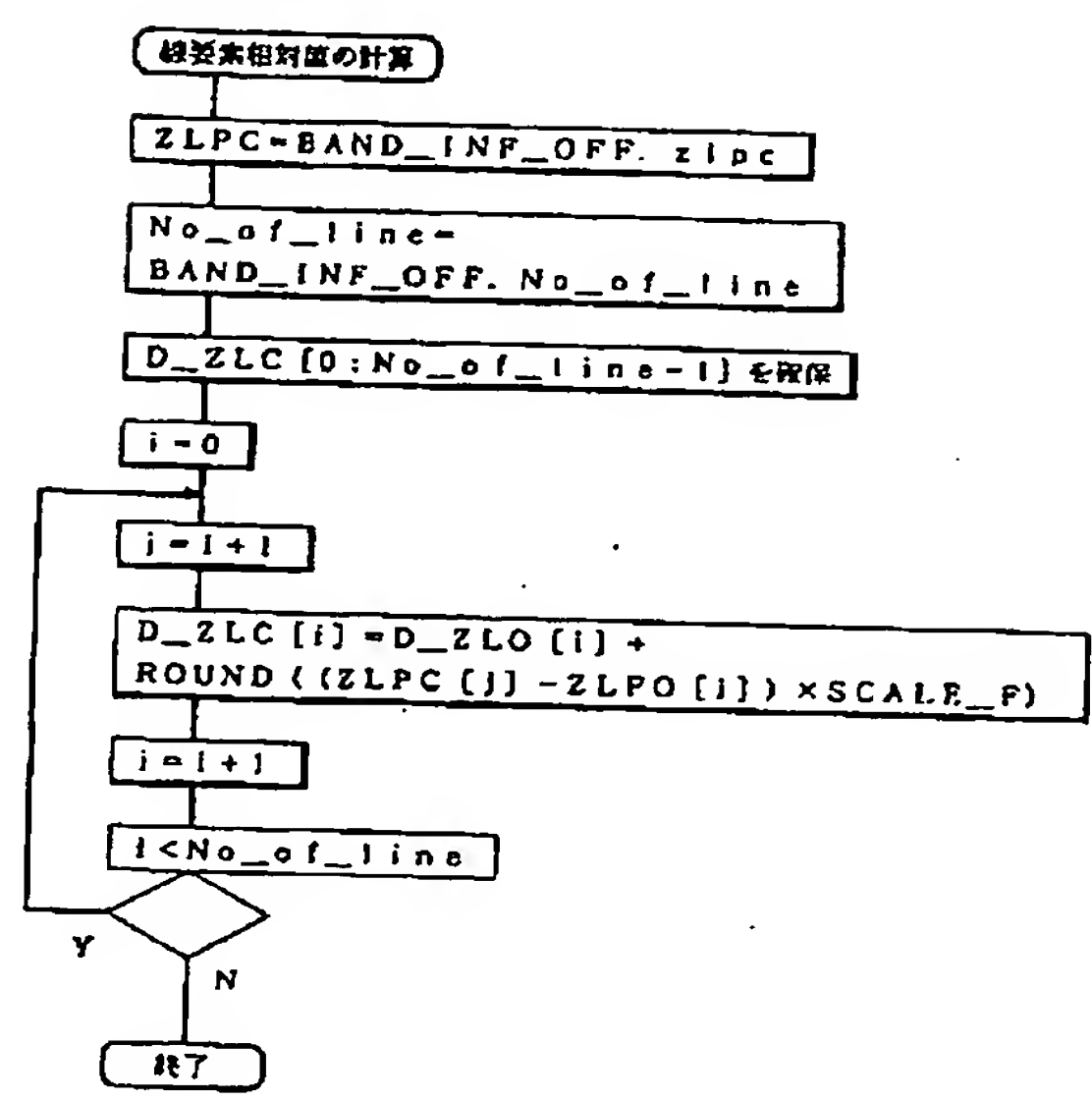
第17図



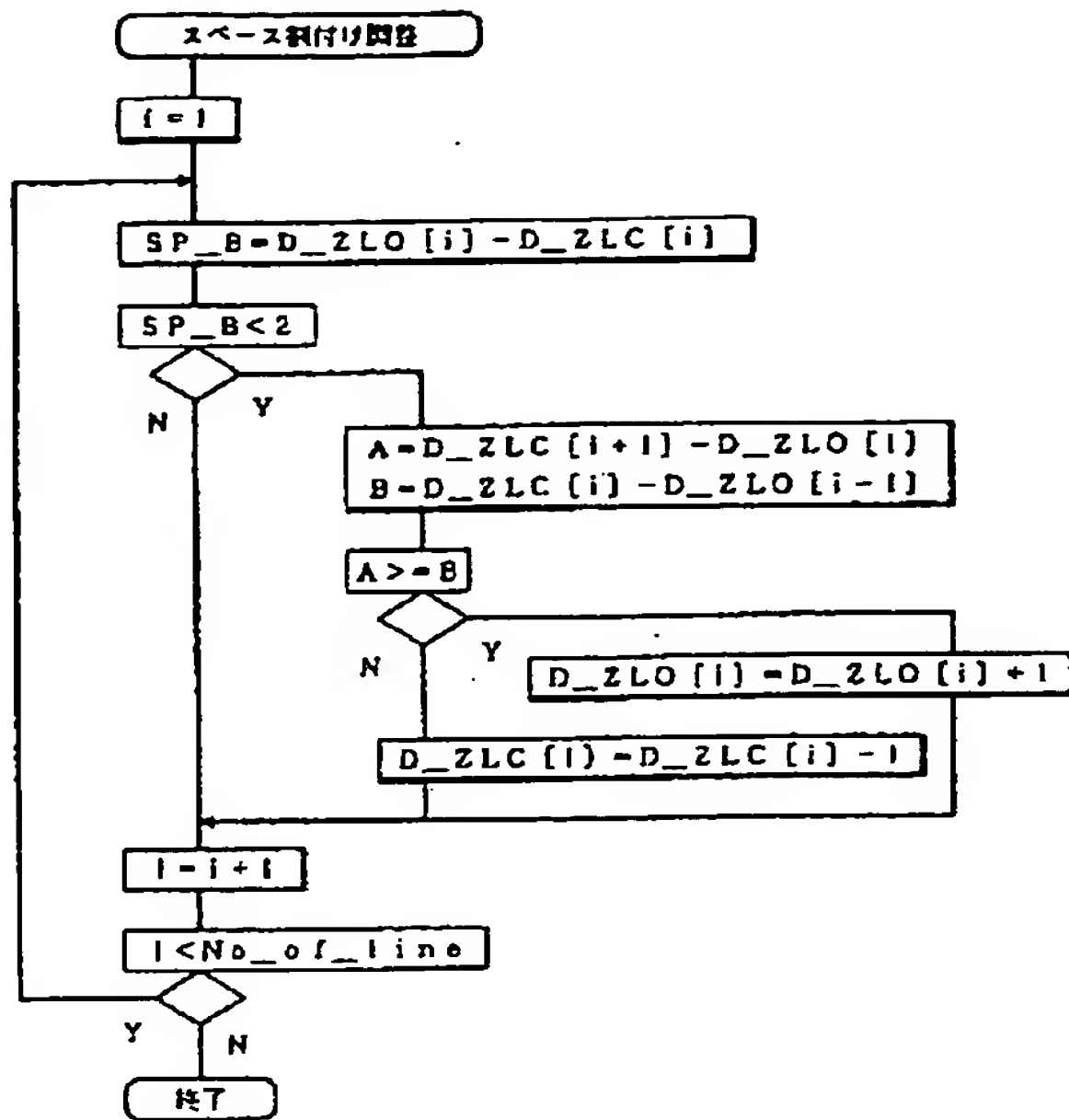
第18図



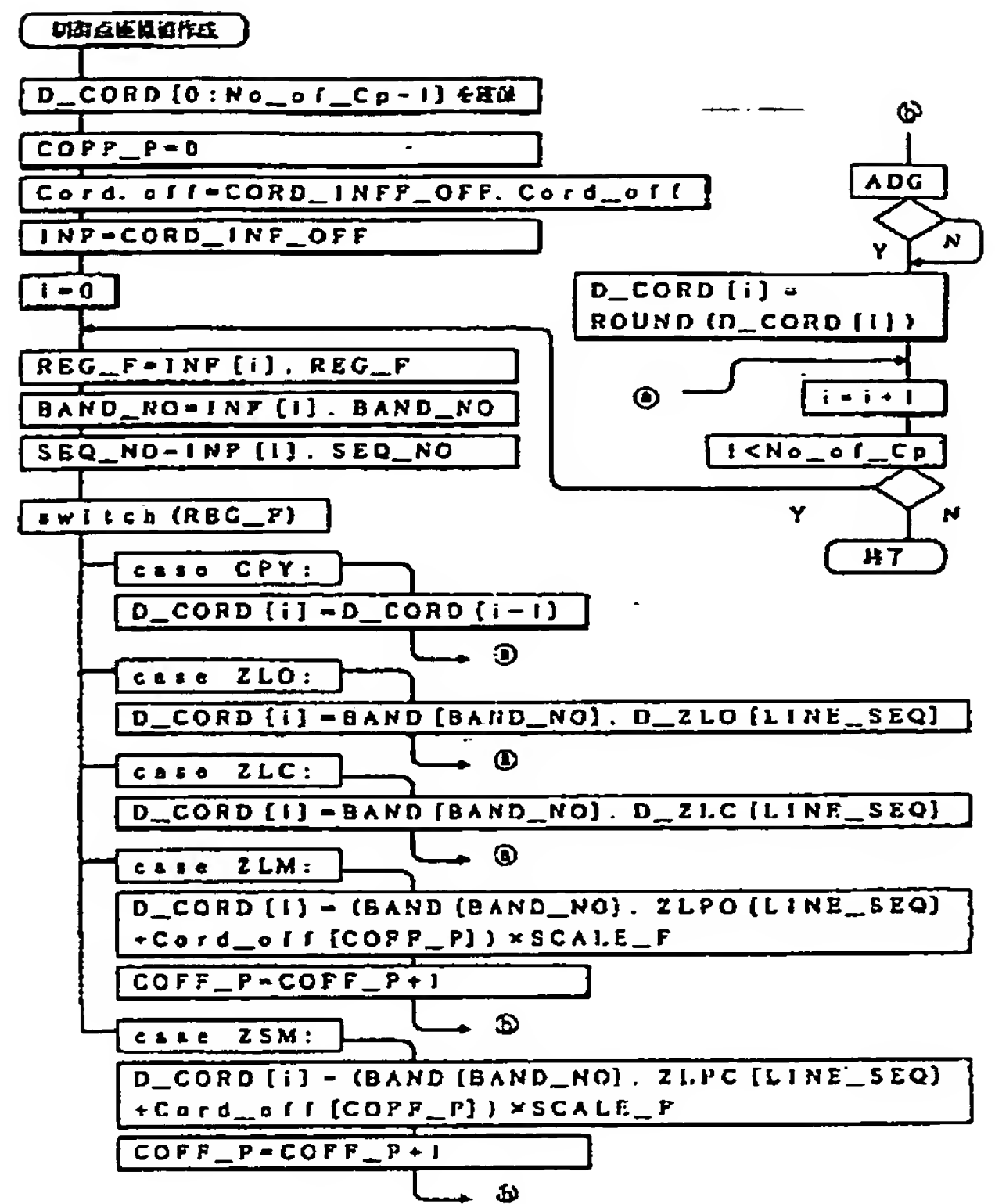
第19図



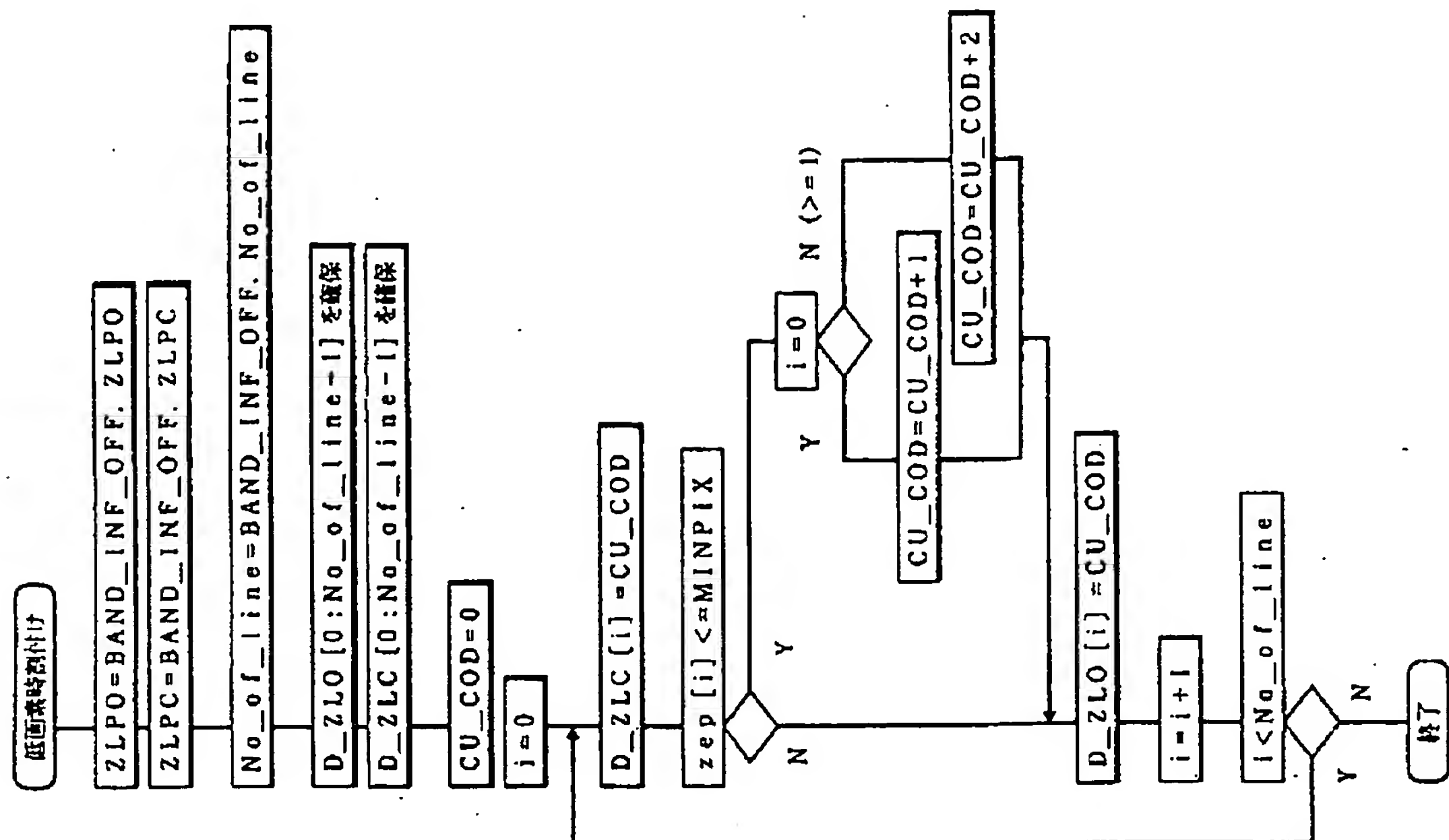
第20図



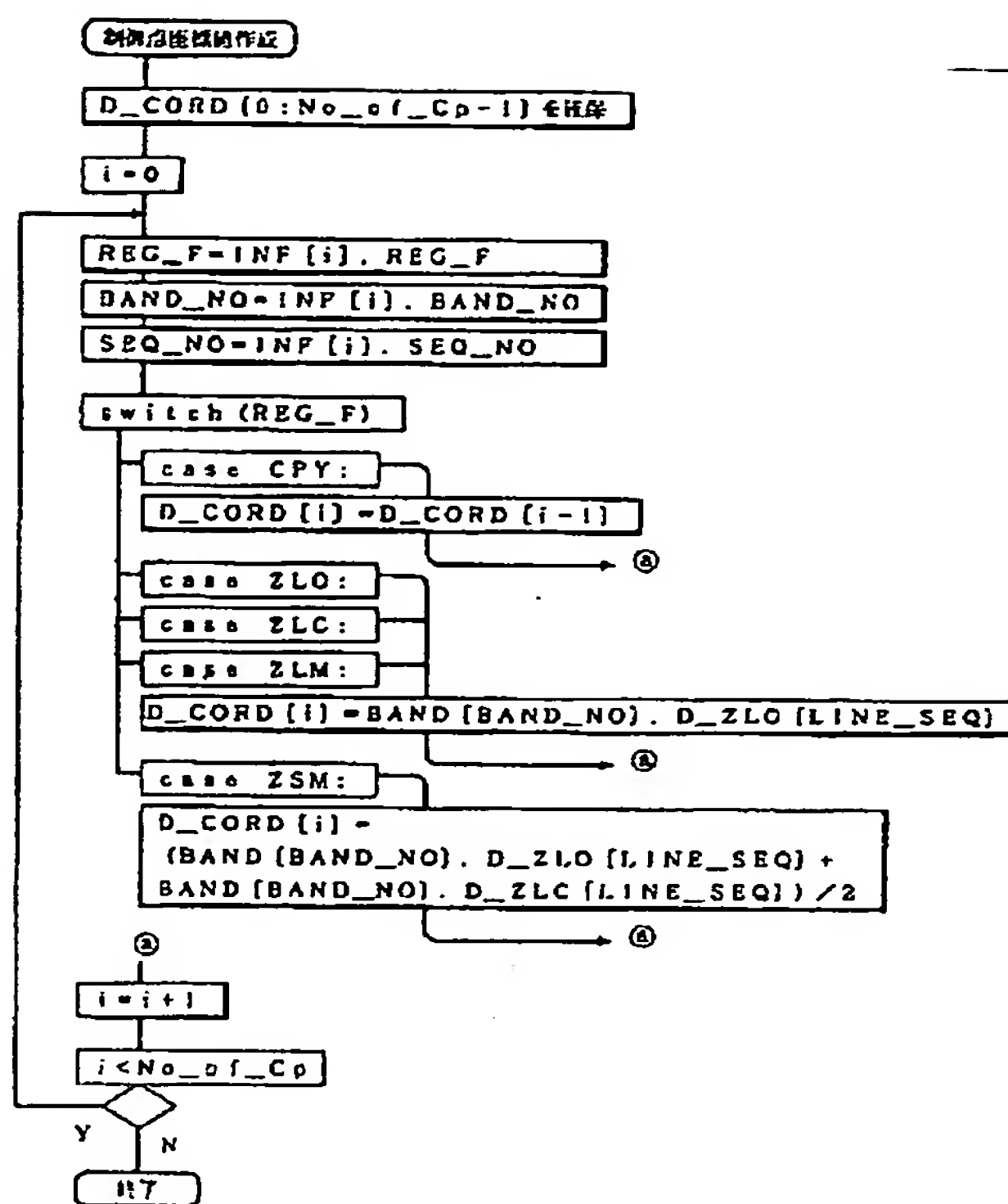
第21図



第22図



第23図



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.